GRAPE: Graph-Augmented Prototype Explanations for Interactive Medical Image Diagnosis

Rasul Khanbayov[1], **Erchin Serpedin**[2], **Hasan Kurban**[1,*]
[1] College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar
[2] Department of Computer and Electrical Engineering,
Texas A&M University, College Station, TX, USA

Abstract

Prototype-based medical image classifiers present three clinical limitations: they treat findings as independent, silently amplify unsafe physician feedback, and require full retraining whenever a new finding is needed. We present GRAPE (Graph-Augmented Prototype Explanations), a unified architecture that targets all three challenges. First, a Graph Attention Task Head boosts macro-F1 by +13.8 pp over the prototype baseline on TBX11K but not on NIH ChestX-ray14, with no graph-specific benefit established. Second, a Concept-Mismatch Safety Check, to our knowledge the first such mechanism in prototype-based medical classifiers, warns when the model's dominant finding inside a doctor-drawn region conflicts with the claimed label, catching 85% of simulated erroneous annotations with no extra inference cost at a favorable operating point, though sensitivity falls substantially at clinically realistic false-positive budgets. Third, Open-Vocabulary Prototype Anchoring aligns visual prototypes to clinical text, allowing a new finding to be added from a single labeled image without modifying any other component, evaluated for localization only. On NIH ChestX-ray14, one Effusion example matched that concept's low fully-supervised Pointing Game ceiling in one run but not in a second. On TBX11K, prototype maps achieve $2.1\times$ better lesion localization than end-to-end concept-based baselines. The graph head adds only $+1$ ms latency at interactive batch size. The project page is available at `https://kurbanintelligencelab.github.io/GRAPE`.

1 Introduction

Deep learning has achieved radiologist-level accuracy on several medical imaging tasks [18, 12], yet clinical adoption remains limited. Two barriers stand out. First, black-box models provide no evidence for their predictions, making it impossible for a physician to verify or override the reasoning. Second, even "interpretable" models may silently propagate spatial errors when doctors provide corrective feedback, creating a patient-safety risk.

Prototype-based models [3, 6] address the first barrier by making predictions through similarity to learned visual examples. Recent work on Concept-based Similarity Reasoning (CSR) [9] extends this to multi-prototype per-concept reasoning with doctor-in-the-loop interaction. However, three structural limitations remain:

1. **Independent concepts.** The task head is a flat linear layer applied to concatenated concept similarity scores. It ignores known anatomical correlations between findings (e.g., active tuberculosis (TB) and cavity formation co-occur reliably), discarding clinically meaningful structure.

*Corresponding author: `hkurban@hbku.edu.qa`

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

2. **Unsafe spatial interaction.** When a physician draws a bounding box to guide the model's attention, the system applies the spatial feedback regardless of the model's confidence in that region. If the model is confused and associates a different concept with the indicated region, the feedback amplifies an error.
3. **Closed concept vocabulary.** Adding a new clinical finding (e.g., a rare variant not seen in training) requires fully retraining the prototype atlas, a prohibitive cost in clinical settings.

We propose **GRAPE**, a unified architecture that targets all three limitations within a coherent four-stage training framework. Our contributions are:

- A **Graph Attention Task Head** (Module A, §3.2) that constructs a data-driven concept co-occurrence graph from training labels and refines prototype similarity scores via a two-layer GAT-style head before classification. In the evaluated implementation, the graph does not affect the head's outputs at inference.
- A **Concept-Mismatch Safety Check** (Module B, §3.3), to our knowledge the first such mechanism in prototype-based medical classifiers that computes per-concept mean similarity inside a physician-drawn region and warns when a different concept dominates, flagging potential error amplification before feedback is applied.
- **Open-Vocabulary Prototype Anchoring** (Module C, §3.4) that aligns visual prototypes to frozen BioViL-T text embeddings, enabling a new clinical concept to be added at test time from a single labeled image without modifying any other component.
- A **four-stage training pipeline** (§3.5) that integrates concept supervision, contrastive prototype learning, VLM alignment, and graph-structured classification.

2 Related Work

ProtoPNet [3] learns class-specific visual prototypes. Extensions add deformability [6], decision trees [16], part-sharing [19], and patch variants [15]. MProtoNet [25] applies prototypes to brain tumor MRI, and Ma et al. [13] visualize prototypical concepts with multiple images. GRAPE builds on CSR [9], replacing its linear head with a graph-structured classifier. Concept Bottleneck Models [10, 7, 27] interpose concept scores before prediction. Margeloiu et al. [14] show they can be semantically misaligned, so GRAPE grounds prototypes to BioViL-T [1] text anchors instead. GCN classifiers [4, 26] model corpus-derived label co-occurrence. GRAPE uses training-label statistics with prototype-similarity node features and learned GAT attention [21]. Bayesian uncertainty [8, 11] requires sampling overhead. GRAPE's prototype-variance signal is deterministic [20]. Medical VLMs (MedCLIP [24], BiomedCLIP [29]) target classification. GRAPE uses text embeddings as geometric anchors for prototype initialization, preserving spatial interpretability. Interactive annotation methods [22] lack concept-mismatch checks. Attribution quality is compared via Insertion/Deletion AUC [17], following the causal deletion/insertion measures it introduces, without implying our own method's maps are shown to be more faithful. B-cos networks [2] offer an alternative faithful attribution approach by replacing standard convolutions with alignment-based transforms. Unlike GRAPE, they do not support interactive concept-level correction or open-vocabulary extension.

3 Method

3.1 Preliminaries: Prototype-Based Concept Reasoning

Backbone F followed by two-layer MLP projector P yields L2-normalized spatial embeddings:

$$\mathbf{f}' = P(\mathbf{f}) \in \mathbb{R}^{D\times h\times w}, \quad \|\mathbf{f}'_{:,i,j}\|_2 = 1 \ \ \forall i, j. \tag{1}$$

Each concept k has M L2-normalized prototypes $\{\mathbf{p}_{km}\} \subset \mathbb{R}^D$. The cosine similarity map $\mathcal{S}_{km}(i,j) = \langle \mathbf{p}_{km}, \mathbf{f}'_{:,i,j} \rangle$ aggregates to a scalar score:

$$s_{km} = \max_{i,j} \mathcal{S}_{km}(i,j). \tag{2}$$

The baseline CSR model feeds the score matrix $\mathbf{s} \in \mathbb{R}^{K\times M}$ through a linear task head to produce class logits. GRAPE replaces this with the three modules described below.

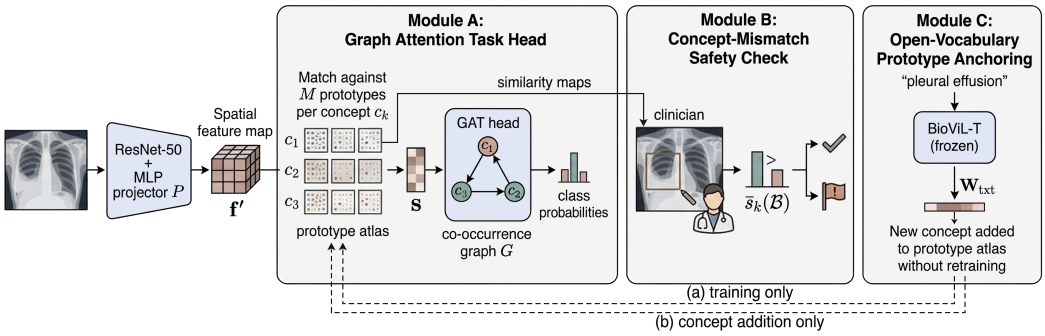

Figure 1: **GRAPE architecture.** An X-ray is encoded by ResNet-50 + MLP projector P into spatial features $\boldsymbol{f}'$. **Module A** matches $\boldsymbol{f}'$ against a learned prototype atlas, and score matrix $\mathbf{s}$ feeds a GAT-style head built over a co-occurrence graph G to produce per-class probabilities. **Module B** compares mean concept similarity $\bar{s}_k(\mathcal{B})$ inside a clinician's box against the claimed concept's, flagging a mismatch before feedback is applied. **Module C** adds concepts from a text anchor via a frozen BioViL-T encoder projected through $\boldsymbol{W}_{\text{txt}}$. Dashed arrows mark training-only (a) or concept-addition-only (b) paths.

3.2 Module A: Graph Attention Task Head

Clinical concepts are not independent. Active TB and cavity formation co-occur, and cardiomegaly correlates with pleural effusion, yet a flat linear head over concatenated s_{km} discards this structure. We build a data-driven co-occurrence graph from training labels $\mathbf{y}^{(n)} \in \{0,1\}^K$. The $K \times K$ conditional probability matrix

$$\mathbf{A}_{kk'} = P(k \mid k') = \frac{\sum_n y_k^{(n)} \cdot y_{k'}^{(n)}}{\sum_n y_{k'}^{(n)} + \epsilon} \tag{3}$$

is thresholded at τ (0.1 on TBX11K, 0.05 on NIH). Edges with $\mathbf{A}_{kk'} > \tau$ keep their weight, self-loops get weight one, and the weights are row-normalized and scale the GAT attention logits. The matrix is *not* symmetrized, so $\mathbf{A}_{kk'} \neq \mathbf{A}_{k'k}$ in general. We denote the resulting graph by G. Since $\mathbf{y}^{(n)}$ is binary, the numerator counts samples where both concepts occur and the denominator counts samples where k' occurs, so $\mathbf{A}_{kk'}$ is the maximum-likelihood estimate of $P(k \mid k')$ [4]. The constant ϵ (denominator clamped to $\min = 1$) prevents division by zero for concepts absent from training.

Graph Attention Network. Each concept is a node with initial feature $\mathbf{h}_k^{(0)} = \mathbf{s}_{k,:} \in \mathbb{R}^M$. Two GAT layers [21] refine these via learned attention (Eqs. 16–18). A LayerNorm–Dropout–Linear readout produces class logits. Layer dimensions and head counts are in Table 7. **Implementation note.** The evaluated head departs from Eq. 18: each node aggregates its own transformed features, so with fixed weights its outputs do not depend on G as long as every node keeps an incoming edge. G affects training only through attention dropout.

3.3 Module B: Concept-Mismatch Safety Check

Prototype variance maps. The disagreement among M prototypes for concept k at patch (i, j) provides a spatial consistency estimate without sampling:

$$\mu_k(i,j) = \frac{1}{M} \sum_{m=1}^{M} \mathcal{S}_{km}(i,j), \tag{4}$$

$$U_k(i,j) = \frac{1}{M} \sum_{m=1}^{M} \left(\mathcal{S}_{km}(i,j) - \mu_k(i,j)\right)^2. \tag{5}$$

High $U_k(i,j)$ indicates that the M prototypes strongly disagree about whether patch (i,j) belongs to concept k, signaling ambiguous evidence. The *safety check* uses the mean map μ_k directly to detect concept mismatch. U_k serves only as an optional gate.

Spatial safety check. When a doctor provides a bounding box $\mathcal{B}$ intending to reinforce concept k^*, GRAPE checks whether the dominant concept inside the region matches. It first computes the mean similarity per concept inside $\mathcal{B}$:

$$\bar{s}_k(\mathcal{B}) = \frac{1}{|\mathcal{B}|} \sum_{(i,j)\in\mathcal{B}} \mu_k(i,j) \quad \forall k, \tag{6}$$

identifies the dominant concept $k^\dagger = \arg\max_k \bar{s}_k(\mathcal{B})$, and issues a safety warning if $k^\dagger \neq k^*$ and $\bar{s}_{k^\dagger}(\mathcal{B}) - \bar{s}_{k^*}(\mathcal{B}) > \eta$ (threshold $\eta = 0.05$). The warning is raised before the feedback is applied. This *concept-mismatch* check is designed to catch cases where the model's dominant response inside the region conflicts with the claimed concept label, before the error can be amplified in the prototype atlas. The rule is uncertainty-free because it compares $\bar{s}_k(\mathcal{B})$ across concepts and never reads U_k. We use the spatial mean rather than the spatial max of μ_k inside $\mathcal{B}$, as the two perform within noise of each other (Table 25).

Optional U_k-gate. U_k (Eq. 5) is used at inference time as an *optional second-stage filter* applied only to boxes the dominance rule already flagged. A flagged mismatch is suppressed if the claimed concept's own per-prototype variance $U_{k^*}(\mathcal{B})$ (the variance across its M prototypes of their mean similarity inside $\mathcal{B}$) exceeds a gate threshold θ_V (distinct from η above). This gate trades true-positive detection for false-positive suppression (§4.5).

3.4 Module C: Open-Vocabulary Prototype Anchoring

We eliminate the closed-vocabulary bottleneck by anchoring each prototype to a clinical text description via the frozen BioViL-T encoder [1]. A learnable projection $\mathbf{W}_{\text{txt}} \in \mathbb{R}^{D\times 768}$ maps each concept's text embedding $\mathbf{t}_k$ into the visual prototype space:

$$\hat{\mathbf{t}}_k = \frac{\mathbf{W}_{\text{txt}}\mathbf{t}_k}{\|\mathbf{W}_{\text{txt}}\mathbf{t}_k\|_2} \in \mathbb{R}^D. \tag{7}$$

The *alignment loss* (Stage 3) pulls all M prototypes toward their text anchor while keeping the BioViL-T backbone frozen:

$$\mathcal{L}_{\text{align}} = \frac{1}{KM} \sum_{k=1}^{K} \sum_{m=1}^{M} \left(1 - \langle \mathbf{p}_{km}, \hat{\mathbf{t}}_k \rangle\right). \tag{8}$$

Zero-shot concept addition. At test time, a clinician adds a new concept k' without image annotations by initializing M perturbed prototypes from the text anchor ($\sigma_{\text{init}} = 0.1$):

$$\mathbf{p}_{k'm}^{(0)} = \frac{\hat{\mathbf{t}}_{k'} + \sigma_{\text{init}}\boldsymbol{\epsilon}_m}{\|\hat{\mathbf{t}}_{k'} + \sigma_{\text{init}}\boldsymbol{\epsilon}_m\|_2}, \quad \boldsymbol{\epsilon}_m \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D),\ \boldsymbol{\epsilon}_m \leftarrow \boldsymbol{\epsilon}_m/\|\boldsymbol{\epsilon}_m\|_2. \tag{9}$$

The perturbation keeps the M prototypes distinct. Without it they would be identical and receive identical gradients during fine-tuning. These prototypes are immediately usable for inference or adaptable with a small number of labeled examples via concept-local fine-tuning.

3.5 Training Pipeline

GRAPE trains in four stages (Algorithm 1): concept supervision, concept vector generation, prototype learning with optional VLM alignment, and GNN task head training.

Stage 1: Concept Supervision. The backbone F and per-concept CAM heads are trained with BCE on Global Average Pooled activations (Eq. 14).

Stage 2: Concept Vector Generation. CAM-weighted spatial averages of frozen backbone features produce per-concept vectors $\mathbf{v}_k$ (Eq. 15) that seed the Stage 3 prototype atlas.

Stage 3: Prototype Learning with VLM Alignment. The projector P, prototypes $\{\mathbf{p}_{km}\}$, and text projection $\mathbf{W}_{\text{txt}}$ are trained jointly. Each projected concept vector $\mathbf{v}'_k = P(\mathbf{v}_k)$ serves as a query, and

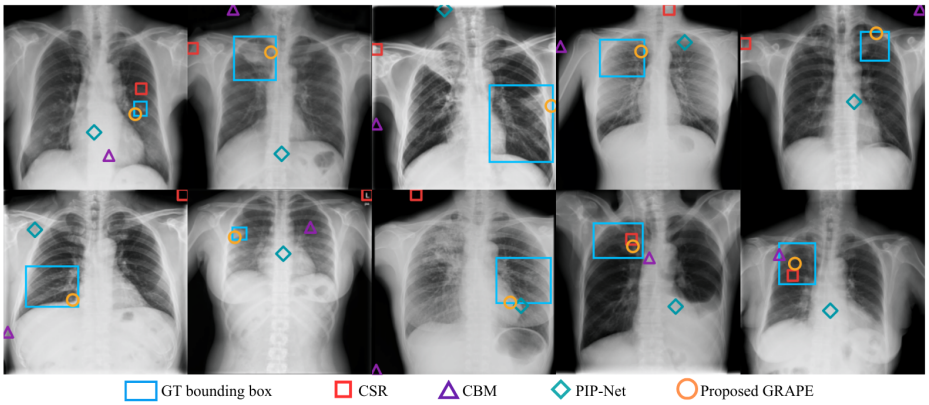

Figure 2: **Pointing Game on TBX11K chest X-rays (Active TB concept).** Each marker shows the max-activation point of a method. In these examples, **GRAPE** (orange circle) localizes the lesion inside the ground-truth bounding box (cyan), while the other methods' maxima mostly fall outside it. These images were selected because **GRAPE**'s point is a hit.

the multi-prototype contrastive loss assigns it to its ground-truth concept $\tilde{k}$ via soft assignment:

$$q_m^k(\mathbf{v}') = \text{softmax}_m(\gamma \langle \mathbf{p}_{km}, \mathbf{v}' \rangle), \tag{10}$$

$$\text{sim}_k(\mathbf{v}') = \sum_m q_m^k(\mathbf{v}') \langle \mathbf{p}_{km}, \mathbf{v}' \rangle, \tag{11}$$

$$\mathcal{L}_{\text{con}} = -\log \frac{\exp\Big(\lambda(\text{sim}_{\tilde{k}}(\mathbf{v}'_{\tilde{k}}) + \delta)\Big)}{\sum_k \exp\Big(\lambda\, \text{sim}_k(\mathbf{v}'_{\tilde{k}})\Big)}, \tag{12}$$

with $\lambda = 10$, $\gamma = 5$, $\delta = 0.1$. As written and implemented, δ enters only the numerator, so it offsets the loss by a constant and does not change its gradients. The Stage 3 loss is $\mathcal{L}_3 = \mathcal{L}_{\text{con}} + 0.1\, \mathcal{L}_{\text{align}}$. Our implementation applies this weight twice, so the effective weight is 0.01.

Stage 4: Graph-Structured Classification. With all prior weights frozen (batch-norm statistics still update), the GNN is trained with $\mathcal{L}_4 = \text{CE}(\text{GNN}(\mathbf{s}, G),\, y)$.

4 Experiments

Tables numbered 4 and above and Figure 4 are in the appendix.

4.1 Datasets

TBX11K [12] is a 3-class TB chest X-ray dataset. We use its 8,400 labeled images (3,800 healthy, 3,800 sick-but-non-TB, and 800 active-TB), with 1,211 bounding box annotations for two TB finding types (Active Tuberculosis, Obsolete Pulmonary Tuberculosis); the third, Pulmonary Tuberculosis, has no boxes or positive labels. We use the official 6,600/1,800 train/val split. Since TBX11K does not release test labels, the 1,800-image validation set also serves as the test split for F1 reporting, and the Stage-4 checkpoint is selected on it. The `bbox_eval` split (200 active-TB images with at least one annotated box) is a subset of the same set and is used for Pointing Game evaluation. All methods share this protocol.

NIH ChestX-ray14 [23] contains 100k+ frontal chest X-rays with 14 finding labels. Primary multi-seed comparison (Table 3) uses a stratified patient-level sub-sample of 10k patients (5 seeds, no patient leakage). Single-seed ablations (Table 2) use a 20k-image shard (seed 42). Pointing Game uses 709 shard images with 888 image–concept pairs (964 boxes) from BBox_List_2017.[2]

Evaluation metrics. Macro F1 measures classification performance averaged across classes, accounting for class imbalance. **Pointing Game (PG)** [28] evaluates spatial interpretability. For each

[2]BBox_List_2017 has 880 annotated images. Boxes were placed on each patient's first streamed image, so NIH Pointing Game and Module B values are approximate.

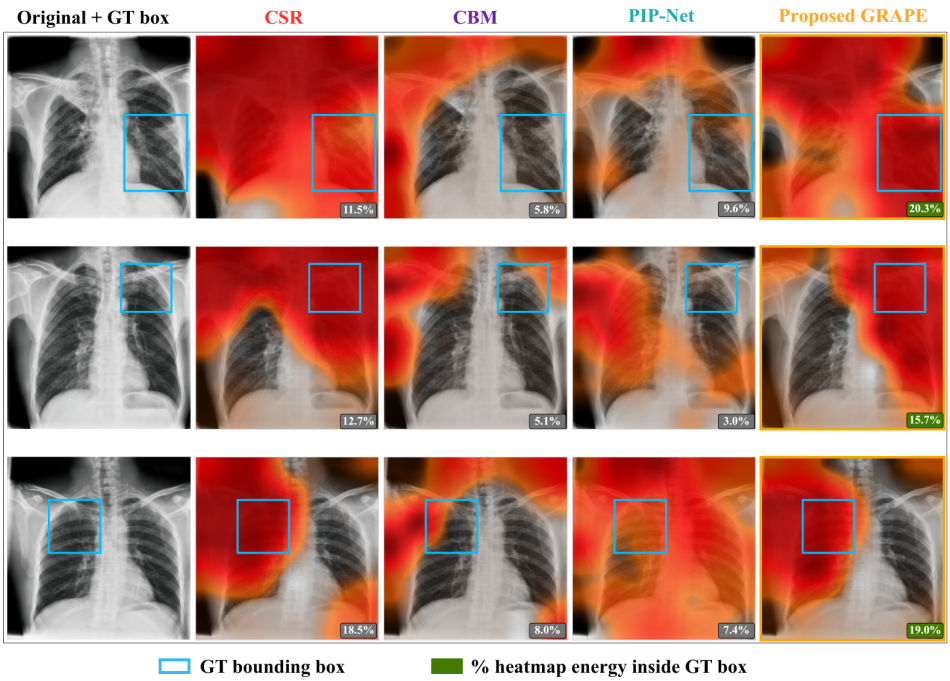

Figure 3: **Prototype similarity maps for CSR Baseline, CBM, PIP-Net, and GRAPE on three Active TB examples.** Heatmaps are overlaid on the input X-rays. Cyan boxes mark ground-truth lesions, and percentage badges give the fraction of total heatmap energy inside the box. Images were chosen to maximize **GRAPE**'s box-energy advantage and are not representative.

(image, concept) pair with a ground-truth bounding box, it checks whether the spatial maximum of the concept's similarity map falls inside the box, and we report the hit rate across all evaluated pairs. Hyperparameters are listed in Tables 7 and 8.

4.2 Main Results

Key observations. **(1) The nonlinear task head is the dominant module on TBX11K.** The GNN head alone saturates the staged gain (+13.9 pp F1), while VLM alone yields no improvement on the sparse $K = 3$ concept set. E2E training reaches 0.990 ± 0.002 F1 but at the cost of localization (PG 0.091 ± 0.059 vs. staged 0.111 ± 0.038, Table 9). A plain ResNet-50 linear head achieves 0.993 F1, confirming E2E gains are driven by unconstrained backbone fine-tuning rather than concept reasoning. The per-class gain is concentrated on majority classes (Healthy $+21.5$ pp, Sick-non-TB $+20.6$ pp, Table 15). Active TB F1 is comparable between GRAPE and CSR, suggesting the GNN reshapes the output distribution rather than improving disease-specific detection.

(2) VLM alignment on NIH. VLM contributes to the single-seed gain (Table 2), but multi-seed evaluation (Table 3) shows GRAPE and CSR are statistically comparable ($p = 0.894$), so the gain does not persist under the patient-level split protocol.

(3) Pointing Game localization. GRAPE's prototype maps achieve $\mathbf{2.1\times}$ **higher PG than CBM** (0.117 vs. 0.055) **and PIPNet** (0.117 vs. 0.053, 3-seed means, Table 1). End-to-end fine-tuning sacrifices prototype-to-region alignment for classification accuracy. On the single-seed run (seed 42), GRAPE achieves PG $= 0.1401$ vs. CSR 0.0628 ($+7.73$ pp, Table 11). Multi-seed evaluation finds no statistically significant difference vs. CSR ($p = 0.64$), reflecting high variance ($\sigma \approx 0.03$) inherent to Pointing Game at 7×7 resolution.

4.3 Graph Head Comparison and Inference Speed

We ablate the choice of graph task head by comparing GRAPE's GAT with ML-GCN [4] and ADD-GCN [26]. All three heads outperform the linear baseline (+10.7–14.6 pp F1) and perform within

Table 1: Ablation on TBX11K (3-class TB detection). We report Macro F1 and Pointing Game (PG); higher is better for both. All results are mean ± std over three seeds $\{0, 1, 42\}$. GRAPE is designed to improve grounding (PG) under a curriculum protocol and is *not* expected to surpass end-to-end baselines on F1 (Observation 1). Best F1 and PG within each block are in **bold**. The CSR macro-F1 (0.706) differs from Tables 3 (0.728), 11 (0.7519), and 27 (0.798) because each uses a different seed count or protocol (see Table 27).

Method	Module			Macro F1 ↑	PG ↑
	GNN	Unc.[§]	VLM		
Module ablation under curriculum training					
CSR[†] [9]	–	–	–	0.706 ± 0.059	0.077 ± 0.016
+ GNN	✓	–	–	0.845 ± 0.049	0.117 ± 0.045
+ VLM	–	–	✓	0.722 ± 0.032	0.097 ± 0.044
+ GNN + VLM	✓	–	✓	$\mathbf{0.875 \pm 0.020}$	0.087 ± 0.033
GRAPE (GNN + Unc.)	✓	✓	–	0.845 ± 0.049	$\mathbf{0.117 \pm 0.045}$
End-to-end variants and concept-based baselines[¶]					
ResNet-50 (backbone only)[•]	–	–	–	$\mathbf{0.993}$	–
GRAPE-E2E[‡]	✓	✓	–	0.9900 ± 0.0020	$\mathbf{0.0910 \pm 0.0590}$
CBM [10]	–	–	–	0.9778 ± 0.0109	0.0550 ± 0.0187
PIPNet [15]	–	–	–	0.9644 ± 0.0021	0.0533 ± 0.0184

[†] CSR re-implemented; no published numbers on these splits. [§] Module B is inference-only; does not affect F1/PG. [‡] E2E: task gradients through backbone throughout. [•] ResNet-50+GAP+linear, seed 42; E2E gains driven by unconstrained fine-tuning. [¶] Same backbone/splits; CBM: joint bottleneck; PIPNet: patch-prototype similarity.

seed variance of one another (Table 11), so the choice of head is not the driver of the gain. We retain the GAT-style head as the evaluated configuration.

Graph structure vs. added nonlinearity. All three graph heads are nonlinear readouts, so that comparison alone cannot separate a relational (graph) effect from a capacity (nonlinearity) effect. To isolate the two, we train a parameter-matched 2-layer MLP (≈47K parameters, Table 27) directly on the flattened $K \times M$ score vector, replacing graph structure with a generic nonlinear function of the same capacity. On TBX11K, the MLP achieves F1 $= 0.865$, closing $\sim 69\%$ of the gap between the linear head (0.798) and the GAT (0.896) on this single seed. The remaining $+3.1$ pp ($0.865 \rightarrow 0.896$) is a difference between readouts, not a graph effect: the sole cross-concept edge is worth 0.25 pp (Table 13), and the evaluated head's outputs do not depend on G at inference. The same comparison on NIH ($K = 14$, 5 seeds, patient-level split, Table 28) gives MLP $= 0.5849 \pm 0.0189$ vs. GAT $= 0.6050 \pm 0.0065$ (+2.01 pp, paired $t(4) = 2.33$, $p = 0.080$). Here the MLP has 188K parameters vs. the GAT's 46K, since its input grows with K. The GAT does not clear the MLP by more than seed variance at conventional significance, so no graph-specific benefit is established on either dataset. The GAT did not lose on any of the 5 seeds and showed substantially lower variance.

We also compare a parameter-matched (≈48K) self-attention head with no predefined adjacency (Table 29). On TBX11K it reaches F1 $= 0.906$, PG $= 0.085$, comparable to the GAT's F1 $= 0.900$, PG $= 0.091$ on the same seed (a separate run from Table 27). A nonlinear readout therefore performs comparably with or without a predefined adjacency, consistent with a nonlinear-readout effect rather than a graph effect at this concept count. A per-edge ablation (Table 13) shows self-loops dominate on TBX11K ($K = 3$), since removing a node's only incoming edge silences it. On NIH ($K = 14$, 105 edges), no graph-specific benefit is established. The GNN head adds only $+1$ ms over the linear head at batch size 1, satisfying real-time interaction requirements (Table 12).

4.4 Comparison with CSR Baseline

Table 3 summarizes GRAPE vs. CSR [9] over five seeds on both datasets (patient-level split for NIH). On TBX11K, GRAPE's F1 ranges 0.798–0.916 across seeds (std = 0.042), and the paired t-test yields $t(4) = 8.03$, $p = 0.0013$.

Table 2: Ablation on NIH ChestX-ray14 (binary No Finding vs. any finding, 14 concepts), seed 42 of the initial development run on a 20k-image streaming shard. Module configurations match those in Table 1. Multi-seed mean ± std on the patient-level split is reported in Table 3; per-metric winner in **bold**.

Method	Module			Macro F1 ↑	PG ↑
	GNN	Unc.	VLM		
CSR [9]	–	–	–	0.6166	**0.0766**
+ GNN + Unc.	✓	✓	–	0.6526	0.0664
+ Unc. + VLM	–	✓	✓	0.5859	0.0743
GRAPE (full)	✓	✓	✓	**0.7138**	0.0642

Table 3: Summary comparison of GRAPE against the CSR baseline under the *same* interpretability-constrained training regime (frozen backbone, concept supervision, staged curriculum). All results are mean ± std over five seeds $\{0, 1, 2, 3, 42\}$, with significance from a paired t-test on per-seed Macro F1. End-to-end fine-tuned methods (CBM, PIPNet) use a different training paradigm and are compared in Table 14. Tables 1 and 2 give full ablations. This table's CSR macro-F1 on TBX11K (0.728, 5-seed average) differs from other protocols' measurements of the same quantity (see Table 27).

Dataset	Method	Macro F1 ↑	PG ↑	ΔF1	p-value
TBX11K	CSR[†]	0.728 ± 0.050	0.094 ± 0.033	–	–
	GRAPE (GNN + Unc.)	$\mathbf{0.866 \pm 0.042}$	0.080 ± 0.032[¶]	**+13.8 pp**	**0.0013**
NIH CXR14	CSR[†]	0.6048 ± 0.0170	–[§]	–	–
	GRAPE (full)	0.6031 ± 0.0174	–[§]	−0.2 pp	0.894 (n.s.)

[†] CSR is re-implemented under our training protocol and hyperparameters; published numbers on these splits are unavailable for direct comparison. [§] NIH bounding-box annotations were unavailable for the multi-seed evaluation run; single-seed PG is reported in Table 2. [¶] Paired t-test on per-seed PG: $t(4) = -0.50$, $p = 0.64$ (n.s.). The development-run single-seed advantage (+7.73 pp, Table 11) does not replicate consistently across seeds (seed-level range: −10.5 to +6.0 pp).

4.5 Module B: Safety Check Evaluation

We simulate miscorrections on the TBX11K `bbox_eval` split (207 image–concept pairs, one box per pair, from 309 boxes). With probability p, a bounding box is reassigned to a wrong concept, and the safety check (the dominance rule following Eq. 6, $\eta = 0.05$) issues a warning when the dominant concept inside the box differs from the claimed one. This is simulated disagreement detection, a controllable proxy for model-clinician disagreement, not demonstrated interactive safety under real user behavior. Bootstrap 95% CIs use $B = 1000$ resamples.

With the U_k-gated variant ($\theta_V = 2\times10^{-4}$) as the default operating point, Module B catches **85.1%** of misdrawn boxes with an FP rate of **16.4%** [12.1%, 20.9%] (Table 18). This operating point is selected by minimizing FP subject to TP ≥ 0.85 on this same split. θ_V is checkpoint-specific and must be re-selected per checkpoint. The ungated baseline (Table 17) achieves higher TP (88–89%) at the cost of a higher FP rate (21.7% [16.4%, 27.1%]). Tightening to $\theta_V = 10^{-4}$ drops FP to 12.6% but TP falls to 78.0%, below the 85% clinical threshold. At clinically realistic false-positive budgets the true-positive rate falls substantially further, to 55.7% TP at 7.7% FP and 44.5% TP at 3.4% FP (Table 18). We use the U_k-gated configuration as the default here and report the low-false-positive regime as an explicit limitation. Table 24 shows that the ungated dominance rule substantially outperforms warnings triggered by MC-Dropout uncertainty (TP 90.5% vs. 51.0%/23.4% at FP 22–25%) while requiring only a single deterministic forward pass. On NIH ($K = 14$), TP remains high (0.921 ± 0.012) but FP rises to 0.731 ± 0.011, possibly due to dense concept overlap and approximate box placement. At low false-positive budgets on NIH, a single global threshold reaches 17.9% TP at 4.6% FP and 20.1% TP at 5.4% FP (Table 26). Module B's sensitivity at clinically realistic false-positive rates is therefore substantially lower on NIH than on TBX11K, consistent with the denser concept overlap at $K = 14$ and suggesting that larger vocabularies need per-concept thresholds, not a single global one.

4.6 Further Analysis: Zero-Shot Concept Addition and Attribution

Module C: Zero-shot concept addition. We evaluate Module C via a leave-one-concept-out experiment on TBX11K ($K = 3$). For each held-out concept, we reinitialize its $M = 100$ prototypes with Eq. 9 and fine-tune them with $n \in \{0, 1, 5, 20\}$ labeled images, measuring Pointing Game on the `bbox_eval` split (Table 20). This run loads the model without its text aligner, so Eq. 9 falls back to a random unit vector. Table 21 compares the BioViL-T text anchor with a random one. With no labeled images, this initialization yields non-zero localization for *active_tb* (PG $= 0.030$), and a single labeled image reaches PG $= 0.068 \pm 0.045$, on average 41% of full supervision (68% CI: 14–69%). With five seeds the few-shot trend is essentially monotonic, with 1-shot (PG $= 0.068$) $<$ 5-shot (PG $= 0.079$) $\approx$ 20-shot (PG $= 0.078$).

We also evaluate on NIH ChestX-ray14 ($K = 14$) with the same fallback, where 3 of 4 target concepts have bbox coverage (Infiltration, Effusion, Atelectasis). Effusion's 1-shot PG matched full supervision in this run (PG $= 0.072$, a low absolute ceiling) but not in a second run, and Infiltration's 20-shot reaches 53% of full supervision (Table 22). Adding a concept is not free. Classification F1 on NIH drops by up to 6.7 pp and dominant-concept assignments flip in 19–46% of other findings' boxes (Appendix M.2), so prototype-level non-interference does not imply decision-level non-interference.

A matched random-anchor control (Table 21) tests whether the BioViL-T text anchor itself, rather than any prototype initialization, is responsible for this localization. The two TBX11K concepts disagree about which anchor wins, and with only three seeds neither result is conclusive. The control therefore does not establish one-example localization above chance (Appendix M.1), or that the text-anchoring mechanism specifically makes it possible.

Similarity map attribution. We compare Insertion and Deletion AUC [17], which measure how quickly predicted class probability changes as pixels ranked by saliency are removed or revealed, for GRAPE and CSR on the 200 active-TB images of the TBX11K test split (Table 23). GRAPE's Deletion AUC (0.954 ± 0.068) and Insertion AUC (0.886 ± 0.148) versus CSR's (0.923 ± 0.104 and 0.873 ± 0.138) differ by less than one standard deviation on both metrics. We report this as comparable attribution behavior, not as evidence that either method's similarity maps are more faithful to the underlying decision. Table 14 gives a contextual comparison with SymFormer, CheXNet, CBM, and PIPNet.

5 Discussion

VLM alignment over-regularizes sparse concept sets ($K = 3$). After Stage 3, all $M = 100$ prototypes cluster tightly near their text anchors (Figure 4, mean prototype–anchor cosine $= 0.995 \pm 0.008$ for TBX11K vs. 0.992 ± 0.007 for NIH), and inter-prototype pairwise cosines are similarly high (0.990 vs. 0.984), confirming that the alignment loss collapses intra-concept prototype diversity in both settings.

Safety implications of the concept-mismatch check. The dominance rule following Eq. 6 does not improve macro-F1 directly. Its value is in flagging potentially harmful interactions. In a deployment scenario where a doctor mistakenly draws a box over a region dominated by a different finding, the safety warning gives the system a chance to flag the inconsistency before the feedback is applied.

Limitations. Pointing Game scores remain low in absolute terms (7–14%) because similarity maps at 7×7 resolution must be compared to bounding boxes annotated in 224×224 space. A finer-resolution backbone is untested, and an in-domain backbone at the same 7×7 resolution is compared in Table 31. Modules A, B, and C all operate on the prototype similarity score matrix rather than backbone features directly, so a stronger in-domain backbone is expected to change the quality of the scores these modules receive rather than what the modules do with them. Module B's true-positive rate falls substantially at clinically realistic false-positive budgets (§4.5). The safety-check evaluation itself is a controllable proxy for model-clinician disagreement, simulated via label corruption. It is not a validated model of real clinician behavior under ambiguous boxes, partial lesions, multi-finding regions, or varying clinician expertise. The deployment-relevant disagreement rate that would determine whether any given false-positive rate is tolerable requires a user study we do not have. Module C is additionally evaluated on NIH ($K = 14$, 3 concepts with bbox coverage). NIH boxes were placed on each patient's first streamed image, so NIH Pointing Game and Module B

values are approximate. Full $K = 14$ leave-one-out and a clinician user study remain future work. Robustness of the static co-occurrence graph to a shifted disease prevalence has not been established, because the evaluated head's outputs do not depend on the graph at inference (Table 30).

Training variance. TBX11K F1 ranges 0.798–0.916 (std $= 0.042$, 5 seeds), reflecting Stage 4 GNN sensitivity on the n=6,600 dataset.

Broader Impact

By making every prediction traceable to a specific visual concept and prototype, GRAPE enables clinicians to verify or override model reasoning rather than accepting opaque outputs. The concept-mismatch safety check is designed to reduce the risk that erroneous clinician feedback silently degrades the prototype atlas, a failure mode that could otherwise go undetected in real deployments. The open-vocabulary extension lowers barriers to adoption in resource-limited settings by allowing new clinical concepts to be introduced without full retraining. At the same time, prototype-based explanations that appear clinically plausible may not be faithful to the model's actual decision boundary, and clinicians who over-rely on similarity maps risk anchoring diagnostic judgment to potentially misleading visualizations. This work does not evaluate performance across demographic subgroups. Prototype atlases learned from imbalanced distributions may exhibit worse spatial alignment for underrepresented populations, and practitioners should audit subgroup performance before deployment.

Ethics statement. This work uses two publicly released datasets: TBX11K [12], released under a research license, and NIH ChestX-ray14 [23], released by the U.S. National Institutes of Health (NIH) under a public-access license. No new human-subject data were collected for this research. All experiments are conducted retrospectively on de-identified images. The system described in this paper is a research prototype and is *not* deployed in any clinical setting.

6 Conclusion

We presented GRAPE, an interactive medical image diagnosis architecture combining concept-mismatch interaction safety (Module B), a nonlinear task head (Module A), and open-vocabulary prototype anchoring (Module C) in a unified four-stage training framework. Module B, to our knowledge the first concept-mismatch safety check in prototype-based medical classifiers, catches 85% of simulated erroneous bounding-box annotations at 16.4% FP rate (U_k-gated, §4.5) using a single forward pass. Module A yields +13.8 pp macro-F1 (0.866 ± 0.042, 5 seeds, $p = 0.0013$) over the prototype baseline on TBX11K. The gain is concentrated on majority classes (Table 15), the Pointing Game advantage does not replicate consistently across seeds ($p = 0.64$, Table 3), and no graph-specific contribution is established on either dataset (§4.3). Module C is, to our knowledge, the first prototype-based mechanism to add a new clinical concept from a *single* labeled image without retraining, evaluated for localization only. On NIH ChestX-ray14, Effusion 1-shot matched the fully-supervised Pointing Game ceiling for that concept in one run (PG $= 0.072$, no model modification, a low absolute ceiling) but not in a second, and on TBX11K, 1-shot recovers 41% of full supervision. This addition is not free. Classification accuracy, calibration, and other concepts' dominant-concept assignments all shift measurably downstream, non-monotonically in the number of labeled examples (§4.6). Insertion/Deletion AUC evaluation (§4.6) finds GRAPE and CSR differ by less than one standard deviation, indicating comparable attribution behavior rather than an advantage for either method. Code and training scripts are available at `https://github.com/KurbanIntelligenceLab/GRAPE`.

Acknowledgments and Disclosure of Funding

The authors thank the TBX11K and NIH ChestX-ray14 dataset providers for making their data publicly available. This work used an NVIDIA A100 80GB GPU and a Tesla P100 16GB GPU.

Funding: The authors received no third-party funding for this work. **Competing Interests:** The authors declare no competing interests.

References

[1] Shruthi Bannur, Stephanie Hyland, Qianchu Liu, Fernando Perez-Garcia, Maximilian Ilse, Daniel C Castro, Benedikt Boecking, Harshita Sharma, Kenza Bouzid, Anja Thieme, et al. Learning to exploit temporal structure for biomedical vision-language processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15016–15027, 2023.

[2] Moritz Böhle, Mario Fritz, and Bernt Schiele. B-cos networks: Alignment is all we need for interpretability. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10319–10328. IEEE, 2022.

[3] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: deep learning for interpretable image recognition. *Advances in Neural Information Processing Systems*, 32, 2019.

[4] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5177–5186, 2019.

[5] Joseph Paul Cohen, Joseph D Viviano, Paul Bertin, Paul Morrison, Parsa Torabian, Matteo Guarrera, Matthew P Lungren, Akshay Chaudhari, Rupert Brooks, Mohammad Hashir, et al. Torchxrayvision: A library of chest x-ray datasets and models. In *International Conference on Medical Imaging with Deep Learning*, pages 231–249. PMLR, 2022.

[6] Jon Donnelly, Alina Jade Barnett, and Chaofan Chen. Deformable protopnet: An interpretable image classifier using deformable prototypes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10265–10275, 2022.

[7] Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Zohreh Shams, Frederic Precioso, Stefano Melacci, Adrian Weller, et al. Concept embedding models: Beyond the accuracy-explainability trade-off. *Advances in Neural Information Processing systems*, 35:21400–21413, 2022.

[8] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pages 1050–1059. PMLR, 2016.

[9] Ta Duc Huy, Sen Kim Tran, Phan Nguyen, Nguyen Hoang Tran, Tran Bao Sam, Anton Van Den Hengel, Zhibin Liao, Johan W Verjans, Minh-Son To, and Vu Minh Hieu Phan. Interactive medical image analysis with concept-based similarity reasoning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 30797–30806, 2025.

[10] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR, 2020.

[11] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, 30, 2017.

[12] Yun Liu, Yu-Huan Wu, Shi-Chen Zhang, Li Liu, Min Wu, and Ming-Ming Cheng. Revisiting computer-aided tuberculosis diagnosis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(4):2316–2332, 2023.

[13] Chiyu Ma, Brandon Zhao, Chaofan Chen, and Cynthia Rudin. This looks like those: Illuminating prototypical concepts using multiple visualizations. *Advances in Neural Information Processing Systems*, 36:39212–39235, 2023.

[14] Andrei Margeloiu, Matthew Ashman, Umang Bhatt, Yanzhi Chen, Mateja Jamnik, and Adrian Weller. Do concept bottleneck models learn as intended? *arXiv preprint arXiv:2105.04289*, 2021.

[15] Meike Nauta, Jörg Schlötterer, Maurice Van Keulen, and Christin Seifert. Pip-net: Patch-based intuitive prototypes for interpretable image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2744–2753, 2023.

[16] Meike Nauta, Ron Van Bree, and Christin Seifert. Neural prototype trees for interpretable fine-grained image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14933–14943, 2021.

[17] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*, 2018.

[18] Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

[19] Dawid Rymarczyk, Łukasz Struski, Jacek Tabor, and Bartosz Zieliński. Protopshare: Prototype sharing for interpretable image classification and similarity discovery. *arXiv preprint arXiv:2011.14340*, 2020.

[20] Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential deep learning to quantify classification uncertainty. *Advances in Neural Information Processing Systems*, 31, 2018.

[21] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[22] Guotai Wang, Wenqi Li, Maria A Zuluaga, Rosalind Pratt, Premal A Patel, Michael Aertsen, Tom Doel, Anna L David, Jan Deprest, Sébastien Ourselin, et al. Interactive medical image segmentation using deep learning with image-specific fine tuning. *IEEE Transactions on Medical Imaging*, 37(7):1562–1573, 2018.

[23] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2097–2106, 2017.

[24] Zifeng Wang, Zhenbang Wu, Dinesh Agarwal, and Jimeng Sun. Medclip: Contrastive learning from unpaired medical images and text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3876–3887, 2022.

[25] Yuanyuan Wei, Roger Tam, and Xiaoying Tang. MProtoNet: A case-based interpretable model for brain tumor classification with 3D multi-parametric magnetic resonance imaging. In *Medical Imaging with Deep Learning*, Nashville, United States, July 2023.

[26] Jin Ye, Junjun He, Xiaojiang Peng, Wenhao Wu, and Yu Qiao. Attention-driven dynamic graph convolutional network for multi-label image recognition. In *European Conference on Computer Vision*, pages 649–665. Springer, 2020.

[27] Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc concept bottleneck models. *arXiv preprint arXiv:2205.15480*, 2022.

[28] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018.

[29] Sheng Zhang, Yanbo Xu, Naoto Usuyama, Hanwen Xu, Jaspreet Bagga, Robert Tinn, Sam Preston, Rajesh Rao, Mu Wei, Naveen Valluri, et al. Biomedclip: a multimodal biomedical foundation model pretrained from fifteen million scientific image-text pairs. *arXiv preprint arXiv:2303.00915*, 2023.

Appendix overview

The supplementary material is organized as follows.

A	Notation table	p. 14
B	Hyperparameter sensitivity (τ and η sweeps)	p. 15
C	End-to-end training ablation	p. 16
D	Dataset details	p. 17
E	Graph structure	p. 17
F	Graph-head comparison: GAT vs. ML-GCN vs. ADD-GCN	p. 18
G	GAT edge-importance ablation	p. 18
H	Contextual comparison with published methods	p. 19
I	Per-class F1 and confusion matrix	p. 20
J	Staged training algorithm	p. 21
K	Mathematical details	p. 22
L	Per-concept Pointing Game results	p. 22
M	Zero-shot concept addition	p. 23
N	Insertion/Deletion AUC evaluation	p. 24
O	MC-Dropout uncertainty comparison	p. 25
P	Module B: mean vs. max spatial aggregation ablation	p. 25
Q	Module B safety check on NIH ChestX-ray14 ($K = 14$)	p. 26
R	Prototype–anchor alignment and inter-prototype dispersion	p. 26
S	GAT vs. MLP head ablation (TBX11K & NIH)	p. 27
T	Self-attention task head ablation	p. 27
U	Co-occurrence graph robustness under distribution shift	p. 28
V	In-domain backbone ablation (ResNet-50 vs. TorchXRayVision)	p. 28

A Notation Table

Table 4: Summary of notation used throughout the paper.

Symbol	Domain	Meaning
Image and features		
$\mathbf{x}$	$\mathbb{R}^{3\times H\times W}$	Input image (H, W: height, width in pixels)
F	—	Backbone network (ResNet-50)
$\mathbf{f}$	$\mathbb{R}^{C_{\text{bb}}\times h\times w}$	Backbone spatial feature map
P	—	Two-layer MLP projector
$\mathbf{f}'$	$\mathbb{R}^{D\times h\times w}$	Projected, L2-normalized patch features
D	$\mathbb{Z}^+$	Prototype embedding dimension (= 256)
Concepts and prototypes		
K	$\mathbb{Z}^+$	Number of concepts
M	$\mathbb{Z}^+$	Prototypes per concept (= 100)
$\mathbf{p}_{km}$	$\mathbb{R}^D$	m-th prototype for concept k (unit-norm)
$\mathcal{S}_{km}(i,j)$	$[-1,1]$	Cosine similarity map: concept k, prototype m, patch (i,j)
s_{km}	$[-1,1]$	Scalar similarity: $\max_{i,j}\mathcal{S}_{km}(i,j)$
$\mathbf{s}$	$\mathbb{R}^{K\times M}$	Score matrix of all scalar similarities
Graph (Module A)		
$\mathbf{A}$	$[0,1]^{K\times K}$	Directed concept co-occurrence matrix ($\mathbf{A}_{kk'}=P(k\mid k')$)
$\mathcal{N}(k)$	—	Neighbourhood of node k in the directed graph
τ	$\mathbb{R}^+$	Co-occurrence threshold (0.1 TBX11K; 0.05 NIH)
F'	$\mathbb{Z}^+$	GAT hidden dimension per head (= 64)
n_h	$\mathbb{Z}^+$	Number of GAT attention heads (Layer 1: 4; Layer 2: 1)
Safety check (Module B)		
$\mu_k(i,j)$	$[-1,1]$	Mean prototype similarity at patch (i,j) for concept k
$U_k(i,j)$	$[0,1]$	Prototype variance (population) at patch (i,j) for concept k
$\mathcal{B}$	—	Doctor-drawn bounding box region (set of patch indices)
$\bar{s}_k(\mathcal{B})$	$[-1,1]$	Mean of μ_k inside box $\mathcal{B}$
k^*	$\{1,\dots,K\}$	Doctor's claimed concept for the drawn box
$k^\dagger$	$\{1,\dots,K\}$	Model's dominant concept inside $\mathcal{B}$
η	$\mathbb{R}^+$	Safety-warning gap threshold (default 0.05)
θ_V	$\mathbb{R}^+$	U_k-gating threshold for FP suppression
B_{boot}	$\mathbb{Z}^+$	Bootstrap resampling count (= 1000)
VLM alignment (Module C)		
$\mathbf{t}_k$	$\mathbb{R}^{768}$	BioViL-T text embedding for concept k
$\mathbf{W}_{\text{txt}}$	$\mathbb{R}^{D\times 768}$	Learnable text-to-visual projection
$\hat{\mathbf{t}}_k$	$\mathbb{R}^D$	Projected, normalized text anchor (unit-norm)
σ_{init}	$\mathbb{R}^+$	Noise scale for zero-shot prototype init (= 0.1)
Training losses		
λ	$\mathbb{R}^+$	Contrastive temperature (= 10)
γ	$\mathbb{R}^+$	Soft-assignment sharpness (= 5)
δ	$\mathbb{R}^+$	Inter-concept margin (= 0.1)
λ_{align}	$\mathbb{R}^+$	VLM alignment loss weight (= 0.1)
$\tilde{k}$	$\{1,\dots,K\}$	Ground-truth concept index for projected vector $\mathbf{v}'_{\tilde{k}}$

B Hyperparameter Sensitivity

Implementation. Backbone: ResNet-50 pretrained on ImageNet at 224×224. Prototype dimension $D = 256$, $M = 100$ prototypes per concept, batch size 128, mixed precision (BF16 on NVIDIA A100 80GB and FP16 on the Tesla P100 16GB, Appendix S).

We report sensitivity to the two most impactful thresholds: the graph co-occurrence threshold τ and the safety check threshold η. All experiments use TBX11K unless noted. Fixed hyperparameters are listed in Table 7.

Graph threshold τ. For each $\tau \in \{0.01, 0.05, 0.10, 0.20, 0.30\}$ we rebuild the co-occurrence graph and retrain only Stage 4, warm-started from the trained head (20 more epochs, LR 10^{-3}, weight decay 10^{-4}). Table 5 reports Macro F1 and Pointing Game on the TBX11K validation set.

Table 5: Sensitivity of GRAPE to the graph co-occurrence threshold τ on TBX11K. Stage 4 (the GAT task head) is warm-started from the trained head for each τ with Stages 1–3 held fixed. The default ($\tau = 0.10$, used in all other TBX11K experiments) is highlighted. Macro F1 varies by less than 0.6 pp across the swept range, as expected since the evaluated head's outputs do not depend on the graph at inference. PG shows more variation (0.135–0.150) but no monotonic trend.

τ	Macro F1 ↑	PG ↑
0.01	**0.9177**	0.1353
0.05	0.9159	0.1353
0.10 (default)	0.9163	**0.1498**
0.20	0.9124	0.1401
0.30	0.9159	0.1401

Safety threshold η. Using the trained GRAPE model (no retraining), we sweep $\eta \in \{0.01, 0.03, 0.05, 0.10, 0.20\}$ in the safety check experiment of §4.5 (miscorrection rate $p = 0.5$, $T = 5$ trials).

Table 6: Sensitivity of the Module B safety check to the threshold η on TBX11K `bbox_eval` ($N = 207$ pairs). True-positive (TP) rate is evaluated at miscorrection rate $p = 0.5$; false-positive (FP) rate at $p = 0$. The default operating point ($\eta = 0.05$, used in all other experiments) is highlighted. Both TP and FP are essentially flat across the swept range, so the safety check is not sensitive to this threshold.

η	TP rate ↑	FP rate ↓
0.01	0.905	0.217
0.03	0.905	0.217
0.05 (default)	**0.903**	**0.217**
0.10	0.901	0.217
0.20	0.895	0.198

TP rate varies by about 1 pp across the sweep, and FP rate is essentially constant (decreasing only at $\eta = 0.20$, where small gaps are ignored). The default $\eta = 0.05$ keeps TP high without inflating FP.

VLM alignment on sparse vs. dense concept sets. TBX11K has only $K = 3$ sparse TB-type concepts with high class imbalance (800 active-TB vs. 3,800 healthy images in train+val). The alignment loss $\mathcal{L}_{\text{align}}$ pulls all 300 prototypes toward three text anchors, which may over-constrain the prototype geometry when the visual diversity is limited and the task signal is dominated by the binary presence/absence of TB lesions.

Table 7: Fixed hyperparameters used across all GRAPE experiments. Stage numbering follows §3. Stage 3 trains the prototype layer with contrastive and alignment objectives, and Stage 4 trains the GAT task head. Per-dataset varying hyperparameters (learning rate, weight decay, epochs) are listed in Table 8 below.

Parameter	Symbol	Value	Stage
Prototype layer			
Prototypes per concept	M	100	all
Prototype dimension	D	256	all
Contrastive scale	λ	10	3
Assignment sharpness	γ	5	3
Inter-concept margin	δ	0.1	3
VLM alignment weight	λ_{align}	0.1	3
Task head (GAT)			
GAT hidden dimension	d_h	64	4
GAT attention heads	n_h	4	4
Label smoothing	ε_{ls}	0.05	4
Optimization			
Batch size	–	128	all

Table 8: Per-dataset varying hyperparameters for GRAPE. Learning rate (LR), weight decay (WD), and epoch counts are tuned per dataset. All other parameters are fixed as in Table 7. All stages use AdamW; Stage 3–4 weight decay is the AdamW default (10^{-2}).

Dataset	Stage	LR	WD	Epochs
TBX11K	Stage 1	1×10^{-4}	1×10^{-4}	30
	Stage 2	– (feature extraction)	–	–
	Stage 3	1×10^{-4}	1×10^{-2}	20
	Stage 4	1×10^{-3}	1×10^{-2}	20
NIH ChestX-ray14	Stage 1	5×10^{-5}	1×10^{-4}	30
	Stage 2	– (feature extraction)	–	–
	Stage 3	5×10^{-5}	1×10^{-2}	20
	Stage 4	5×10^{-4}	1×10^{-2}	20

C End-to-End Training Ablation

To examine the role of the staged curriculum, we train GRAPE (GNN + Uncertainty, no VLM) with all parameters jointly from scratch for 70 epochs, the same total epoch budget as staged training (30 + 20 + 20). No staged objectives are used. The loss is $\mathcal{L} = \mathcal{L}_{\text{BCE}} + \mathcal{L}_{\text{proto}} + \mathcal{L}_{\text{cls}}$ throughout, where $\mathcal{L}_{\text{proto}}$ is the multi-prototype contrastive loss from Eq. 12 (see §3.5), and the backbone receives task-classification gradients from epoch 1. We run both strategies over three seeds $\{42, 0, 1\}$ to compare variance.

Table 9: Curriculum ablation on TBX11K. We compare GRAPE's staged curriculum (Stages 1–4 trained sequentially with frozen earlier-stage weights) against end-to-end joint training of all stages, with all other hyperparameters held fixed. Mean $\pm$ std over three seeds $\{0, 1, 42\}$. E2E Pointing Game (PG) is per-image (a hit if any concept hits). Per-metric winner in **bold**. See §4.2 for the F1–PG trade-off.

Training strategy	Macro F1 ↑	PG ↑
Staged curriculum (GRAPE)	0.809 ± 0.084	$\mathbf{0.111 \pm 0.038}$
End-to-end (no curriculum)	$\mathbf{0.990 \pm 0.002}$	0.091 ± 0.059

Result. E2E achieves substantially higher and more stable F1 (0.990 ± 0.002) compared with staged training (0.809 ± 0.084), confirming that the ResNet-50 backbone is the bottleneck in staged training. However, *staged training achieves higher mean Pointing Game accuracy* (0.111 ± 0.038 vs. 0.091 ± 0.059) with lower variance.

Table 10: Zero-shot and few-shot concept addition (Module C, leave-one-out) on TBX11K for the *active_tb* concept, comparing staged curriculum training against end-to-end (E2E) joint training. Pointing Game (PG) is reported on `bbox_eval` pairs. Staged values (*active_tb*) are from an earlier three-seed run of the Table 20 protocol, matching the E2E seed count. E2E values are means over $3 \times 3 = 9$ runs (3 backbone training seeds $\times$ 3 few-shot adaptation seeds). Per-row winner in **bold**.

Labeled examples (n)	Staged PG ↑	E2E PG ↑
0 (no examples)	0.030	$\mathbf{0.062 \pm 0.025}$
1	$\mathbf{0.079 \pm 0.054}$	0.064 ± 0.016
5	0.063 ± 0.037	$\mathbf{0.078 \pm 0.036}$
20	0.071 ± 0.025	$\mathbf{0.084 \pm 0.049}$

Zero-shot comparison. Table 10 shows that E2E weights also support Module C's zero-shot initialization: the random-anchor fallback produces non-zero localization at $n = 0$ (PG $= 0.062$, higher than staged 0.030). However, E2E zero-shot PG improves only marginally as n increases ($+0.022$ from $n = 0$ to $n = 20$), while staged training shows a steeper gain from $n = 0$ to $n = 1$ ($+0.049$). We attribute this to prototype entanglement. In E2E training, backbone and prototypes co-evolve, making prototype geometry less amenable to targeted few-shot fine-tuning after training.

When to use each. Despite higher F1, we retain the staged curriculum as the primary configuration for three reasons:

1. **Modular interpretability.** Staged training decouples concept learning, prototype learning, and task classification, and each stage can be inspected, retrained, or replaced independently.
2. **Localization.** Staged training achieves higher and more stable PG (0.111 vs. 0.091).
3. **Few-shot responsiveness.** Staged prototypes respond more cleanly to few-shot fine-tuning for new concepts, a property required for Module C in clinical deployment.

We recommend E2E training for practitioners where classification accuracy is the primary goal and interpretability and zero-shot addition are not required.

D Dataset Details

TBX11K concept descriptions (used for VLM alignment):

- *ActiveTuberculosis*: "Active pulmonary tuberculosis with cavitary lesions, nodular opacities, and upper-lobe infiltrates, often with signs of consolidation on chest X-ray."
- *ObsoletePulmonaryTuberculosis*: "Healed or inactive tuberculosis with calcified granulomas, fibrous scarring, and volume loss in the upper lobes on chest X-ray."
- *PulmonaryTuberculosis*: "Pulmonary tuberculosis presenting as patchy or confluent consolidation, tree-in-bud opacities, or miliary nodules on chest X-ray."

Bounding box scaling. TBX11K COCO annotations store boxes in original image coordinates (variable resolution). We scale to the 224×224 input space as $x_1' = \lfloor x_1 / W_{\text{orig}} \times 224 \rfloor$. NIH bounding box annotations are provided in 1024×1024 coordinates and scaled with the same formula.

E Graph Structure

On TBX11K ($K = 3$), the co-occurrence graph has 3 nodes and 4 edges (3 self-loops and one co-occurrence edge; threshold $\tau = 0.1$). On NIH ($K = 14$), the graph has 14 nodes and 105 edges at threshold $\tau = 0.05$. Concepts like Infiltration–Consolidation and Effusion–Atelectasis form strong edges.

F Graph Head Comparison: GAT vs ML-GCN vs ADD-GCN

GRAPE uses a GAT-style task head [21]. Table 11 ablates this choice by replacing the GAT with two established graph classification baselines: ML-GCN [4] (standard 2-layer GCN with symmetric normalisation) and ADD-GCN [26] (static co-occurrence graph combined with a per-sample dynamic cosine-similarity graph). All three heads receive the same input (prototype similarity scores from frozen Stage 1–3 weights) and are trained for 20 epochs (Stage 4 only).

Table 11: Comparison of graph-based task heads on TBX11K. All graph heads share an identical frozen backbone, projector, and prototype layer (Stages 1–3), and only the Stage 4 task head differs; the linear row is the separate full CSR run, so its Stages 1–3 differ. Graph-based heads use the same label co-occurrence graph (threshold $\tau = 0.10$). Per-metric winner in **bold**. All graph heads substantially outperform the linear baseline on F1. The head cannot change similarity maps, so PG gaps are not head effects. This table selects the head and cannot separate graph structure from nonlinearity (see Table 27). Seed 42, single run. The linear-head F1 differs from Tables 1, 3, and 27 by protocol (see Table 27).

Task head	Macro F1 ↑	PG ↑	ΔF1 (pp)
Linear (CSR baseline) [9]	0.7519	0.0628	–
ML-GCN [4]	**0.8983**	0.1401	+14.6
ADD-GCN [26]	0.8589	**0.1449**	+10.7
GAT (GRAPE)	0.8963	0.1401	+14.4

All three graph heads substantially outperform the linear baseline (+10.7–14.6 pp F1) and perform within seed variance of one another, so the choice of head is not the driver of the gain. ML-GCN achieves marginally the highest F1 (0.8983 vs. 0.8963 for GAT), but the difference is within the seed variance range (std ≈ 0.04, 5-seed estimate) observed in Table 3 and is not statistically significant. ADD-GCN has slightly lower F1 (0.8589), and its higher PG (0.1449) is not a head effect. We retain the GAT-style head as GRAPE's evaluated configuration. Its attention weights do not affect the outputs (§3.2).

Table 12: Inference latency comparison between the GNN head and a linear head, measured on a single NVIDIA A100 (BF16) with 200 warmup and 500 timed runs per configuration. At batch size 1, the GNN head incurs only ∼1 ms of additional latency, making it suitable for interactive deployment.

Batch	Head	Mean (ms) ↓	Std (ms)	P95 (ms) ↓	μs/image ↓	Overhead
1	Linear	5.96	0.34	6.23	5958	–
	GNN	6.99	0.26	7.16	6993	+17%
8	Linear	6.05	0.18	6.15	756	–
	GNN	12.17	0.62	12.74	1521	+101%
32	Linear	11.58	0.05	11.65	362	–
	GNN	30.20	0.88	30.90	944	+161%

G GAT Edge-Importance Ablation

The TBX11K co-occurrence graph ($K = 3$, threshold $\tau = 0.10$) has four edges: three self-loops (one per concept) and one directed cross-concept edge $(0 \to 1)$ connecting *ActiveTuberculosis* to *ObsoletePulmonaryTuberculosis*. To measure the contribution of each edge to the task head, we retrain Stage 4 four times, each time removing a single edge from the graph while keeping Stages 1–3 frozen.

Table 13: GAT edge-importance ablation on TBX11K (seed 42). Each row removes one edge from the 4-edge co-occurrence graph and continues training the trained head (warm start). ΔF1 = $\text{F1}_{\text{ablated}} - \text{F1}_{\text{full}}$.

Removed edge	Type	Macro F1	ΔF1 (pp)
None (full graph)	—	0.9188	—
$0\to0$ (self-loop, ActiveTB)	self	0.8539	-6.49
$2\to2$ (self-loop, PulmonaryTB)	self	0.8989	-1.99
$1\to1$ (self-loop, ObsoleteTB)	self	0.9140	-0.48
$0\to1$ (ActiveTB $\to$ ObsoleteTB)	cross	0.9162	-0.25

The self-loop on concept 0 (ActiveTuberculosis) is the most informative edge. Removing it causes the largest drop of -6.49 pp F1, indicating that the GAT head relies heavily on self-aggregation of the dominant positive class. The PulmonaryTB self-loop contributes moderately (-1.99 pp), while the ObsoleteTB self-loop and the sole cross-concept edge $(0\to1)$ each contribute fewer than 0.5 pp, as expected since the evaluated head's outputs do not depend on cross-concept edges at inference.

H Contextual Comparison with Published Methods

On TBX11K, CBM and PIPNet achieve higher raw F1 than GRAPE's staged configuration owing to end-to-end backbone fine-tuning (Observation 1 in §4.2). GRAPE's primary advantage is its localization ($2.1\times$ higher 3-seed PG) and unique capabilities (safety check, zero-shot concept addition). On NIH, the multi-seed evaluation (Table 3) finds GRAPE and CSR statistically comparable ($p = 0.894$). The primary claim rests on TBX11K ($+13.8$ pp F1, $p = 0.0013$). The single-seed PG gain of $+7.73$ pp is not significant over multiple seeds ($p = 0.64$). Table 14 contextualizes GRAPE against published methods. Because each prior work uses a different train/test split, label set, or task formulation, direct numerical comparison is not always valid. The "Comparable" column indicates the degree of comparability.

Table 14: Contextual comparison with published methods on TBX11K and NIH ChestX-ray14. We report the metric used in each method's original paper to avoid lossy reinterpretation, so values are not all directly comparable and each row's comparability is annotated. TBX11K CSR and GRAPE values are single-seed (Table 11). Within comparable rows, GRAPE is benchmarked against CSR (its direct predecessor) under our re-implementation. Concept-based methods (CBM, PIPNet) achieve higher F1 via end-to-end fine-tuning, at the cost of weaker grounding (Table 1).

Method	Split	Metric	Value	Comparable
TBX11K (3-class TB detection)				
Liu et al. [12], SymFormer (P2T-Small)	TBX11K	AUC (TB)	0.991	no[§]
CSR[†] [9]	TBX11K	Macro F1	0.7519	[‡]
GRAPE (GNN + Unc.)	TBX11K	Macro F1	**0.8963**	[‡]
CBM [10]	TBX11K	Macro F1	0.9778 ± 0.0109	[‡]
PIPNet [15]	TBX11K	Macro F1	0.9644 ± 0.0021	[‡]
NIH ChestX-ray14 (binary No Finding vs. any finding, 14 concepts)				
CheXNet [18]	NIH CXR14	Mean AUC	0.841	partial[§]
CSR[†] [9]	NIH CXR14	Macro F1	0.6048 ± 0.0170	[‡]
GRAPE (full)	NIH CXR14	Macro F1	0.6031 ± 0.0174	[‡]

[†] Re-implemented under our training protocol; published numbers on these splits are unavailable for direct comparison.
[‡] Reported on our TBX11K split (§4.1) or the NIH patient-level test split with comparable evaluation protocol. [§] Reported on a different split or label subset; included for context only and not directly comparable to the metrics in this table.

On TBX11K, CBM and PIPNet achieve higher raw F1 (0.9778 and 0.9644) than GRAPE's staged configuration (0.8963, seed 42), because both baselines train end-to-end without the prototype-preserving curriculum. Liu et al.'s SymFormer (AUC 0.991) uses a different metric. On NIH,

CheXNet (mean AUC 0.841 over 14 findings) uses a different split and task formulation and is listed for context.

I Per-Class F1 and Confusion Matrix

Table 15 reports per-class macro-F1 for GRAPE and CSR across five seeds on TBX11K, revealing the source of the macro-F1 gap. Table 16 shows the confusion matrix for seed 42.

Table 15: Per-class F1 on the TBX11K test set, mean $\pm$ std over five seeds $\{0, 1, 2, 3, 42\}$. GRAPE (GNN + Unc.) is compared against the CSR baseline [9]. GRAPE substantially improves F1 on the two clinically confusable classes (Healthy and Sick non-TB) while maintaining performance on Active TB. Per-class winner in **bold**.

Method	Healthy ↑	Sick non-TB ↑	Active TB ↑
CSR [9]	0.628 ± 0.097	0.622 ± 0.063	$\mathbf{0.932 \pm 0.010}$
GRAPE	$\mathbf{0.843 \pm 0.050}$	$\mathbf{0.828 \pm 0.056}$	0.927 ± 0.024

GRAPE's macro-F1 improvement is driven almost entirely by the two majority classes: Healthy ($+0.215$ mean) and Sick-non-TB ($+0.206$ mean). Active TB F1 is already high in both models (≈ 0.93) and barely changes. This pattern is not a co-occurrence effect, since the evaluated head's outputs do not depend on the graph at inference. CSR's high Active-TB score but low Healthy/Sick scores reflect the known class-imbalance pattern in TBX11K (800 active-TB vs. 3,800 healthy and 3,800 sick images).

Table 16: Confusion matrices on the TBX11K test set (seed 42). Rows index the ground-truth class and columns the predicted class. Diagonal entries (correct predictions) are shown in **bold**. GRAPE substantially reduces Healthy $\leftrightarrow$ Sick non-TB confusion relative to CSR ($75+103 = 178$ vs. $213+315 = 528$ off-diagonal errors in this 2×2 block) while maintaining comparable Active TB recall.

		Predicted			Recall
	Ground truth	Healthy	Sick non-TB	Active TB	
GRAPE					
GT	Healthy	**724**	75	1	90.5%
GT	Sick non-TB	103	**688**	9	86.0%
GT	Active TB	7	11	**182**	91.0%
CSR baseline [9]					
GT	Healthy	**585**	213	2	73.1%
GT	Sick non-TB	315	**477**	8	59.6%
GT	Active TB	2	14	**184**	92.0%

J Staged Training Algorithm

Algorithm 1 GRAPE: four-stage training pipeline.

Require: Training set $\mathcal{D} = \{(\mathbf{x}^{(n)}, \mathbf{y}^{(n)}, y^{(n)})\}_{n=1}^{N}$; concept descriptions $\{d_k\}_{k=1}^{K}$; hyperparameters $\lambda_{\text{align}}, M$
Ensure: Trained parameters $\theta = \{F, P, \{\mathbf{p}_{km}\}, \mathbf{W}_{\text{txt}}, \phi_{\text{GNN}}\}$
// Stage 1: Concept supervision.
Train backbone F and per-concept CAM heads end-to-end:
  $\mathcal{L}_1 = \frac{1}{K}\sum_{k=1}^{K} \text{BCE}(\text{GAP}(\text{CAM}_k(F(\mathbf{x}))),\, y_k)$
// Stage 2: Concept vector extraction (backbone frozen).
for each $(\mathbf{x}, \mathbf{y}) \in \mathcal{D}$ **and** concept $k \in \{1, \ldots, K\}$ **do**
  $\mathbf{v}_k \leftarrow \sum_{i,j} \text{softmax}_{ij}(\text{CAM}_k(i,j)) \cdot \mathbf{f}_{:,i,j}$
// Stage 3: Prototype learning (train P, $\{\mathbf{p}_{km}\}$, $\mathbf{W}_{txt}$).
Encode and normalize text descriptions: $\hat{\mathbf{t}}_k \leftarrow \mathbf{W}_{\text{txt}}\mathbf{t}_k / \|\mathbf{W}_{\text{txt}}\mathbf{t}_k\|_2$
Initialize $\{\mathbf{p}_{km}\}_{m=1}^{M}$ by random selection from $\{\mathbf{v}_k^{(n)}\}_{n=1}^{N}$
Minimize $\mathcal{L}_3 = \mathcal{L}_{\text{con}} + \lambda_{\text{align}}\,\mathcal{L}_{\text{align}}$
// Stage 4: Graph classification (all earlier stages frozen).
Construct concept co-occurrence graph G from training labels (Eq. 3)
Compute score matrix $\mathbf{s} \in \mathbb{R}^{K \times M}$ from frozen prototypes
Train graph head ϕ_{GNN}: $\mathcal{L}_4 = \text{CE}(\phi_{\text{GNN}}(\mathbf{s}, G),\, y)$

Table 17: Module B safety check on TBX11K `bbox_eval` ($N = 207$ pairs, threshold $\eta = 0.05$), **ungated** dominance rule. We report true-positive (TP) and false-positive (FP) rates as a function of the miscorrection rate p, the simulated probability that the user's correction is incorrect, as bootstrap means with 95% CIs ($B = 1000$). TP is undefined at $p = 0$ (no misdrawn boxes) and FP is undefined at $p = 1$ (no correct boxes). The other dashes mark values not measured in this sweep. A later re-run measuring both rates at every p found FP 25.6–26.6% and TP 80.2–87.2%. TP is stable across p, so detection does not depend on miscorrection prevalence. Table 18 reports the default U_k-gated operating point.

Miscorrection rate p	TP rate ↑ [95% CI]	FP rate ↓ [95% CI]
0.00	–	0.217 [0.164, 0.271]
0.10	0.884 [0.722, 1.000]	–
0.25	0.885 [0.788, 0.964]	–
0.50	0.886 [0.822, 0.941]	–
1.00	0.887 [0.841, 0.928]	–

Table 18: Module B safety check, U_k**-gated** operating points on TBX11K `bbox_eval` ($N = 207$ pairs, $\eta = 0.05$), swept over the false-positive-suppression threshold θ_V. The gate is a second-stage filter applied only to boxes the ungated dominance rule already flagged (Table 17). If the claimed concept's own per-prototype similarity variance U_{k^*} exceeds θ_V, the flagged mismatch is suppressed. The default setting ($\theta_V = 2{\times}10^{-4}$) selects the threshold that minimizes FP subject to TP ≥ 0.85. The low-false-positive rows ($\theta_V \in \{0.0006, 0.0008\}$) show the trade-off at clinically realistic FP budgets. These two rows come from a later re-run on a different checkpoint (ungated FP 25.6%), so their θ_V values are not comparable with the other rows.

θ_V	TP rate ↑	FP rate ↓	Note
0.0006	0.445	0.034	
0.0008	0.557	0.077	
$1{\times}10^{-4}$	0.780	0.126	below 85% TP clinical floor
$2{\times}10^{-4}$	**0.851**	**0.164** [0.121, 0.209]	**default**
none (ungated)	0.887	0.217 [0.164, 0.271]	Table 17, p=1.00/p=0.00 rows

K Mathematical Details

This appendix collects the standard formulas referenced in the main text.

Cosine similarity map (§3.1). Given L2-normalised patch features $\mathbf{f}'_{:,i,j} \in \mathbb{R}^D$ and prototype $\mathbf{p}_{km} \in \mathbb{R}^D$, the similarity map is:

$$\mathcal{S}_{km}(i,j) = \langle \mathbf{p}_{km},\, \mathbf{f}'_{:,i,j} \rangle. \tag{13}$$

Because both vectors are L2-normalised, this inner product is equivalent to cosine similarity.

Stage 1 concept supervision loss (§3.5). The backbone and per-concept CAM head are trained with binary cross-entropy on Global Average Pooled activations:

$$\mathcal{L}_1 = \frac{1}{K} \sum_{k=1}^{K} \mathrm{BCE}(\mathrm{GAP}(\mathrm{CAM}_k(\mathbf{f})),\, y_k)\,. \tag{14}$$

This is a standard multi-label BCE objective applied to per-concept binary labels.

Stage 2 CAM-weighted concept vector (§3.5). For each training image, the local concept vector is the CAM-attention-weighted spatial average of backbone features:

$$\mathbf{v}_k = \sum_{i,j} \mathrm{softmax}_{ij}(\mathrm{CAM}_k(i,j)) \cdot \mathbf{f}_{:,i,j}. \tag{15}$$

This is the standard CAM aggregation formula, and the resulting vectors seed the initialization of Stage 3 prototypes.

GAT message-passing equations (§3.2). Standard Veličković et al. attention over the concept co-occurrence graph. For layer ℓ and head h, the unnormalised attention score, normalised weight, and updated node representation are:

$$e^h_{kk'} = \mathrm{LeakyReLU}\Big(\mathbf{a}_h^\top \Big[\mathbf{W}^h_{\mathrm{GAT}} \mathbf{h}_k^{(\ell)} \,\|\, \mathbf{W}^h_{\mathrm{GAT}} \mathbf{h}_{k'}^{(\ell)}\Big]\Big), \quad k' \in \mathcal{N}(k), \tag{16}$$

$$\alpha^h_{kk'} = \frac{\exp(e^h_{kk'})}{\sum_{j \in \mathcal{N}(k)} \exp(e^h_{kj})}, \tag{17}$$

$$\mathbf{h}_k^{(\ell+1)} = \mathrm{ELU}\left(\sum_{k' \in \mathcal{N}(k)} \alpha^h_{kk'}\, \mathbf{W}^h_{\mathrm{GAT}} \mathbf{h}_{k'}^{(\ell)} \right). \tag{18}$$

L Per-Concept Pointing Game Table

Table 19: Per-concept Pointing Game (PG) on TBX11K `bbox_eval` for GRAPE (GNN + Unc. configuration). Per-pair PG counts each annotated (image, concept) pair as one trial. We additionally report the per-image PG, where an image is counted as a hit if any concept's maximum activation falls within its bounding box. The two aggregation rules give numerically similar results (0.1353 per-pair vs. 0.1400 per-image) but are computed differently and should not be conflated. *pulmonary_tuberculosis* has no `bbox_eval` pairs and is excluded.

Concept	Pairs	PG (per-pair) ↑	Hits
active_tb	164	0.1646	27
obsolete_tb	43	0.0233	1
Total (per-pair, weighted)	207	0.1353	28
Total (per-image)	200	0.1400	28

M Zero-Shot Concept Addition Results

We perform a leave-one-concept-out experiment on TBX11K ($K = 3$). For each held-out concept k', we load the fully trained GRAPE model, reinitialize only concept k''s $M = 100$ prototype vectors using the initialization of Eq. 9 ($\sigma_{\text{init}} = 0.1$), and fine-tune those prototypes alone with $n \in \{0, 1, 5, 20\}$ labeled images while keeping all other parameters frozen. This run loads the model without its text aligner, so Eq. 9 falls back to a random unit vector in place of the BioViL-T anchor. Appendix M.1 compares the two directly. CSR's closed vocabulary cannot add a concept at test time, so there is no CSR baseline here.

Table 20: Zero-shot and few-shot concept addition (Module C) on TBX11K under a leave-one-concept-out proxy with $K = 3$ concepts. We report Pointing Game (PG) accuracy on `bbox_eval` pairs for the two evaluable concepts, and *pulmonary_tuberculosis* is excluded as it has no bounding-box pairs. "Full supervision" denotes the original GRAPE model trained on all examples for the held-out concept. Mean $\pm$ std over five seeds for $n > 0$; $n = 0$ is a single run (random-anchor fallback, no labeled examples).

Labeled examples (n)	*active_tb* PG↑	*obsolete_tb* PG↑
0 (no examples)	0.030	**0.093**
1	0.068 ± 0.045	0.047 ± 0.036
5	0.079 ± 0.025	0.005 ± 0.009
20	0.078 ± 0.072	0.023 ± 0.029
Full supervision	**0.165**	0.023

For both concepts, the $n = 0$ result reflects random prototype initialization. Without the text aligner the anchor falls back to a random unit vector (§4.6), so these values do not represent semantic grounding and should not be compared against the few-shot rows. With labeled examples, *obsolete_tb* performance is inconsistent: 1-shot reaches PG $= 0.047 \pm 0.036$ but drops sharply at $n = 5$ (0.005 ± 0.009), recovering to 0.023 ± 0.029 at $n = 20$. Full supervision for *obsolete_tb* is itself very low (PG $= 0.023$, 43 pairs), so the concept is intrinsically hard to localize; the high variance across seeds reflects this difficulty.

M.1 Random-Anchor Control

Table 20 uses the random-unit-vector fallback, so it does not establish what the BioViL-T text anchor contributes over a random initialization. The difference between the two concepts' n=0 values above comes from the same fallback, so it settles nothing. We therefore run a matched control. For each held-out concept, we repeat the leave-one-concept-out protocol twice, once from the BioViL-T text anchor and once from a random unit vector, with the same seeds, the same $\sigma_{\text{init}} = 0.1$ perturbation, and the same fine-tuning schedule at each n. The anchor is the only difference between the two runs.

Table 21: Random-anchor control for Module C on TBX11K (3 seeds for n>0, $\sigma_{\text{init}} = 0.1$, identical protocol to Table 20 otherwise). This is a separate run, so absolute values differ from that table. *pulmonary_tuberculosis* is excluded as before (no bbox-eval pairs).

	active_tuberculosis PG↑		*obsolete_pulmonary_tb* PG↑	
n	Text anchor	Random anchor	Text anchor	Random anchor
0	0.085	0.098	0.047	**0.000**
1	0.079 ± 0.005	0.091 ± 0.026	0.031 ± 0.011	0.016 ± 0.011
5	0.069 ± 0.008	$\mathbf{0.108 \pm 0.021}$	$\mathbf{0.054 \pm 0.011}$	0.039 ± 0.011
20	$\mathbf{0.085 \pm 0.005}$	0.069 ± 0.020	0.039 ± 0.011	0.031 ± 0.011

The two concepts disagree about which anchor wins, and with only three seeds neither result is conclusive. On *active_tuberculosis*, the random anchor exceeds the text anchor at $n \leq 5$, most clearly at n=5 (0.108 vs. 0.069), and the text anchor wins at n=20 (0.085 vs. 0.069). On *obsolete_pulmonary_tb*,

the text anchor wins at every n, including n=0, where the random anchor produces zero hits. On this evidence, whether the text anchor helps over a random vector depends on the concept. Module C's one-example results (Table 20) are near the chance level of uniform-random Pointing Game on this grid (0.063 for *active_tuberculosis*, 0.048 for *obsolete_pulmonary_tb*; 0.073 for Effusion on NIH), and they are not evidence that the text-anchoring mechanism specifically makes localization possible.

M.2 NIH ChestX-ray14 Partial Leave-One-Out

We extend the Module C evaluation to NIH ChestX-ray14 ($K = 14$), targeting four concepts: Infiltration (113 pairs), Effusion (138 pairs), Atelectasis (166 pairs), and Consolidation. Consolidation has no bounding-box annotations in NIH ChestX-ray14 and is excluded. For each of the three evaluable concepts, we load the fully trained NIH GRAPE model (seed 0), reinitialize that concept's $M = 100$ prototypes with Eq. 9 (random-unit-vector fallback, as in Table 20, since this run also loads the model without its text aligner), and fine-tune with $n \in \{0, 1, 5, 20\}$ labeled images over 3 seeds.

Table 22: Zero-shot and few-shot concept addition (Module C) on NIH ChestX-ray14 under a partial leave-one-out evaluation ($K = 14$, 3 of 4 target concepts have bbox coverage). Pointing Game (PG) on `bbox_eval` images. Mean $\pm$ std over 3 seeds for $n > 0$; $n = 0$ is a single run (random-anchor fallback). *Effusion* $n = 1$ std $= 0.000$: all three seeds converged identically.

Labeled examples (n)	*Infiltration* PG ↑	*Effusion* PG ↑	*Atelectasis* PG ↑
0 (no examples)	0.044	0.029	0.042
1	0.021 ± 0.015	$\mathbf{0.072}$	0.012 ± 0.009
5	0.050 ± 0.071	0.046 ± 0.007	0.006 ± 0.009
20	$\mathbf{0.094 \pm 0.063}$	0.039 ± 0.025	$\mathbf{0.050 \pm 0.020}$
Full supervision	$\mathbf{0.177}$	0.072	0.048

The NIH few-shot results are mixed. *Effusion* matches its full-supervision PG at $n = 1$ (0.072), though this ceiling is itself low. *Infiltration* reaches 53% of full supervision at $n = 20$ (0.094 vs. 0.177), and *Atelectasis* at $n = 20$ (0.050) marginally exceeds its fully supervised PG (0.048, 166 pairs). Trends at low n remain noisy. Infiltration and Atelectasis dip at $n = 1$ or $n = 5$ before recovering at $n = 20$, while Effusion peaks at $n = 1$. The high variance at $n = 5$ for Infiltration (± 0.071) reflects sensitivity to the specific images selected by each seed.

Interference from adding a concept. Adding a concept is not free. We re-add Effusion to a separately trained NIH model (seed 42, BioViL-T anchor, 3 adaptation seeds). In this run full-supervision PG is 0.051 and 1-shot PG is 0.039, so the 1-shot match in Table 22 does not reproduce here. Classification F1 falls from 0.602 to 0.578, 0.535, and 0.587 at $n = 1, 5, 20$, and ECE rises from 0.094 to 0.118, 0.152, and 0.099, so the cost is non-monotonic in n. The mean F1 of the other 13 concepts changes by at most 0.2 pp, but the dominant concept inside other findings' boxes changes for 19–46% of boxes once $n \geq 1$ (Atelectasis: 37% at $n = 1$). Adding Hernia, which has almost no co-occurrence edge to Atelectasis, flips Atelectasis boxes at the same rate (39% at $n = 1$), so the flips are not specific to graph-adjacent concepts. Existing prototypes cannot move by construction, so prototype-level non-interference does not imply decision-level non-interference.

N Insertion/Deletion AUC Evaluation

We compare attribution behavior using Insertion and Deletion AUC [17] over the 200 active-TB images of the TBX11K test split, using $\text{sal}(i, j) = \max_{k,m} \mathbf{S}_{km}(i, j)$ as the saliency signal. Deletion AUC measures how quickly predicted class probability drops as salient pixels are replaced by channel mean (lower is better). Insertion AUC measures how quickly it rises as salient pixels are revealed from a Gaussian-blurred baseline (higher is better).

GRAPE's higher Deletion AUC (+3.1 pp) indicates that removing the maximally salient patches does not fully disrupt prediction. We attribute this to the prototype atlas distributing evidence across $M = 100$ spatial prototypes per concept. Each prototype contributes weakly to the saliency aggregate $\max_{k,m} \mathbf{S}_{km}(i, j)$, so the saliency map is more diffuse than a single-activation method. Both models

Table 23: Insertion/Deletion AUC of prototype similarity maps on the 200 active-TB images of the TBX11K test split. Saliency is taken as $\max_{k,m}$ over per-prototype similarity maps. Lower Deletion AUC and higher Insertion AUC indicate better attribution scores. Bold marks the better mean per metric. Differences fall well within one standard deviation, indicating comparable attribution behavior rather than a clear advantage for either method. Mean $\pm$ std across the test set.

Method	Deletion AUC $\downarrow$	Insertion AUC $\uparrow$
CSR [9]	$\mathbf{0.923 \pm 0.104}$	0.873 ± 0.138
GRAPE	0.954 ± 0.068	$\mathbf{0.886 \pm 0.148}$

maintain high confidence throughout the deletion curve (AUC > 0.92), reflecting ResNet-50's global texture encoding. The Insertion curve provides the more informative signal in this regime [17].

O MC-Dropout Uncertainty Comparison

A natural alternative signal for Module B's warnings is MC-Dropout uncertainty [8]: run $T = 30$ stochastic forward passes with dropout enabled at inference, then use either the variance of similarity scores across passes (Variant A) or the predictive entropy (Variant B) to trigger warnings. Table 24 reports TP and FP rates for all three methods, each evaluated at the operating point on its threshold sweep closest to the ungated dominance rule's FP rate ($\approx 21.7\%$).

Table 24: Comparison of warning signals for the Module B safety check on TBX11K `bbox_eval` (207 image pairs, $T_{\text{eval}} = 5$ evaluation trials, miscorrection rate $p = 0.5$). For each method we report the true-positive rate (TP) at the threshold closest to the ungated dominance rule's false-positive (FP) rate of $\approx 21.7\%$ on a fine-grained sweep. Higher TP at matched FP is better. Mean $\pm$ std reported over T_{eval} trials. FP variance is zero because thresholds are calibrated on the same evaluation set.

Signal	TP rate $\uparrow$	FP rate	MC passes $\downarrow$
Dominance rule (GRAPE, ungated)	$\mathbf{0.905 \pm 0.026}$	0.217	**1**
MC-Dropout, score variance	0.510 ± 0.035	0.251	30
MC-Dropout, predictive entropy	0.234 ± 0.049	0.227	30

FP-rate matching. The MC-Dropout score-variance variant slightly overshoots the target FP rate (0.251 vs. 0.217) because its score variance is highly threshold-sensitive: across thresholds in $[10^{-4},\ 5{\times}10^{-4}]$, the FP rate drops from 25.1% to 0.5%, leaving no threshold that exactly matches the dominance-rule operating point. The reported configuration is the closest match on a fine-grained sweep.

The ungated dominance rule substantially outperforms both MC-Dropout variants: Variant A achieves TP $= 51.0\%$ at FP $= 25.1\%$ with a highly threshold-sensitive signal, while Variant B achieves only 23.4% TP because global entropy is weakly correlated with local spatial misdraw. GRAPE's dominance rule reaches 90.5% TP at the same FP level using a single deterministic forward pass rather than $T = 30$ stochastic passes.

P Module B: Mean vs. Max Spatial Aggregation Ablation

The safety check (§3.3) computes $\bar{s}_k(\mathcal{B})$ as the spatial mean of μ_k inside the doctor-drawn box, rather than the spatial maximum. To validate this choice, we evaluate an alternative *max-based* gate: replace $\bar{s}_k(\mathcal{B}) = \frac{1}{|\mathcal{B}|} \sum_{(i,j)\in\mathcal{B}} \mu_k(i,j)$ with $\bar{s}_k^{\max}(\mathcal{B}) = \max_{(i,j)\in\mathcal{B}} \mu_k(i,j)$ in the dominant-concept comparison, keeping all other parameters ($\eta = 0.05$, $T = 5$ trials) identical.

The two variants produce nearly identical results. Mean aggregation yields marginally higher TP (0.895 vs. 0.891) at a marginally higher FP rate (0.217 vs. 0.213). Both differences are within trial noise, so the choice of mean aggregation is not critical.

Table 25: Module B: mean vs. max spatial aggregation on TBX11K bbox_eval (one checkpoint, $T = 5$ trials, $\eta = 0.05$).

Aggregation	TP Rate	FP Rate
Mean (default)	0.895 ± 0.019	0.217
Max	0.891 ± 0.018	0.213

Q Module B Safety Check on NIH ChestX-ray14 ($K = 14$)

We repeat the simulated miscorrection study on the NIH bbox split (888 image–concept pairs, 709 images, 8 concepts with annotations). With $K = 14$ overlapping concepts, the safety check achieves TP $= 0.921 \pm 0.012$, comparable to the TBX11K rate, but the FP rate rises to 0.731 ± 0.011 (vs. 21.7% ungated on TBX11K). The elevated FP rate may reflect the dense concept landscape and approximate box placement. When a correct box is drawn for any single finding, one of the other 13 active concepts often achieves higher mean similarity inside the region (e.g., Pneumonia FP $= 0.90$, Cardiomegaly FP $= 0.85$). High sensitivity is preserved in the multi-label setting, while specificity degrades as concept count and co-occurrence density increase. A learned or per-concept threshold would be needed to deploy Module B in $K>3$ settings.

Table 26: Module B low-false-positive operating points on NIH ($K = 14$), single global threshold, inference-only sweep over the trained checkpoint (no retraining).

Threshold	FP rate ↓	TP rate ↑
4.5×10^{-5}	4.6%	17.9%
5.0×10^{-5}	5.4%	20.1%
none (ungated, re-run)	74.66%	93.31%

Interpretation. At a 5% false-positive budget, a single global threshold on NIH recovers only $\sim$20% true-positive detection, far below both the ungated baseline and TBX11K's low-FP operating points (Table 18: 44.5–55.7% TP at similar FP budgets). Module B's sensitivity at clinically realistic false-positive rates degrades sharply as concept count grows, consistent with the denser concept overlap at $K = 14$. A single global threshold is likely insufficient for deployment at this vocabulary size, and per-concept thresholds remain untested.

R Prototype–Anchor Alignment and Inter-Prototype Dispersion

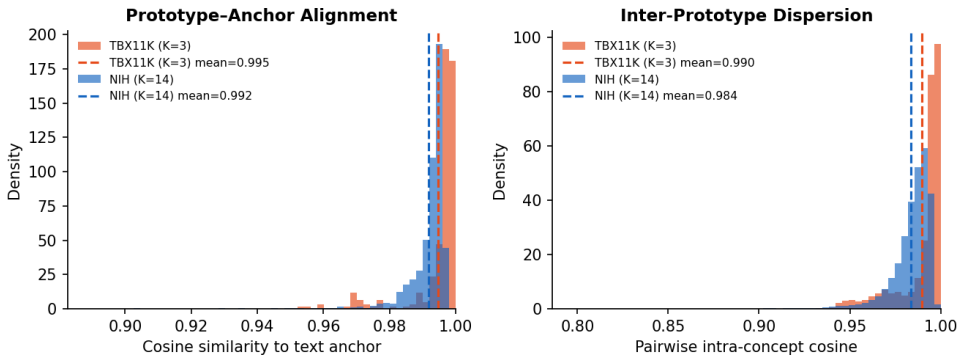

Figure 4: **Prototype geometry after Stage-3 VLM alignment.** *Left:* distribution of cosine similarities between each prototype $\mathbf{p}_{km}$ and its text anchor $\hat{\mathbf{t}}_k$ for TBX11K ($K = 3$, orange) and NIH ($K = 14$, blue). Both distributions are concentrated near 1, confirming that the alignment loss ($\mathcal{L}_{\text{align}}$) pulls prototypes close to their anchors in both settings. *Right:* pairwise cosine similarities between prototypes *within* the same concept ($M = 100$; $\binom{100}{2} = 4{,}950$ pairs per concept). High intra-concept cosines indicate that the M prototypes collapse toward the shared text anchor, reducing intra-concept diversity.

S GAT vs. MLP Head Ablation

Table 27: Ablation isolating the contribution of *graph structure* vs. *added nonlinearity* for the task head on TBX11K. A 2-layer MLP on the flattened $K \times M = 300$ score vector is parameter-matched to the GAT ($\approx$47K vs. $\approx$44K params), trained with the same Stage 4 protocol. The linear head uses the original CSR parameters (903 params). All heads share identical Stage 1–3 weights. Seed 42 single run. **Numeric note:** this table's linear-head F1 (0.798) differs from Table 1 (0.706, 3-seed curriculum protocol), Table 3 (0.728, 5-seed), and Table 11 (0.7519, single-seed task-head-isolation protocol) because each measures the same quantity under a different protocol (seed count, or which Stage-4 run). PG values vary across the same tables for the same reason. PG here is *active_tb* only (164 pairs).

Head	Macro F1 ↑	PG ↑	Params
Linear (CSR)	0.798	0.165	903
MLP (matched)	0.865	0.171	47,235
GAT (GRAPE)	**0.896**	0.165	43,587

Interpretation. The MLP closes $\sim$69% of the Linear→GAT F1 gap (+6.7 pp out of +9.8 pp total) on this single run. The remaining +3.1 pp (MLP→GAT) is a difference between readouts, not a graph effect, because the evaluated head's outputs do not depend on the graph at inference. PG is identical between Linear and GAT, and slightly higher for the MLP.

We repeat the protocol on NIH ChestX-ray14 ($K = 14$, 105 edges at $\tau = 0.05$, 5 seeds {0,1,2,3,42}, patient-level split), retraining Stage 4 from identical frozen Stage 1–3 weights for each seed. The MLP's input grows with K, so on NIH it has 188K parameters vs. the GAT's 46K and is not parameter-matched. *Hardware note.* This NIH ablation and the experiments in Appendices M.1, T, U, and V were run on a single NVIDIA Tesla P100 16GB (batch size 32 on NIH), not the A100 80GB used for the main experiments. We do not expect the choice of GPU to affect results beyond ordinary training-run variance.

Table 28: GAT vs. MLP (188K parameters) on NIH ($K = 14$), 5 seeds, patient-level split. Per-seed macro-F1 and paired t-test.

Seed	0	1	2	3	42
Linear	0.5978	0.5608	0.5910	0.5958	0.6030
MLP (188K)	0.5995	0.5518	0.5914	0.5917	0.5901
GAT (GRAPE)	0.6054	0.6040	0.5984	0.6157	0.6016

Head	Mean ± std	Δ vs. MLP	Paired $t(4)$, p
Linear	0.5897 ± 0.0167	–	–
MLP (188K)	0.5849 ± 0.0189	–	–
GAT (GRAPE)	0.6050 ± 0.0065	+2.01 pp	$t(4) = 2.33$, $p = 0.080$

Interpretation. At conventional significance ($\alpha = 0.05$), the GAT does not clear the MLP by more than seed variance on NIH. No graph-specific benefit is established on either dataset, because the evaluated head's outputs do not depend on the graph at inference. The GAT did not lose to the MLP on any of the 5 seeds, and its seed-to-seed variance (std = 0.0065) is substantially lower than the MLP's (std = 0.0189).

T Self-Attention Task Head Ablation

We test whether a Transformer-style self-attention head over concept nodes, which requires no predefined adjacency, outperforms the GAT. We train a parameter-matched ($\approx$48K params) single-layer multi-head self-attention head directly on the prototype score matrix. Each concept is embedded to $d_{\text{model}} = 96$, attends freely over all K concepts (4 heads, no graph), and a linear readout produces class logits, replacing the GAT's fixed, training-label-derived adjacency with a fully data-driven one learned jointly with the classifier. The protocol otherwise matches Table 27, with frozen Stage 1–3 weights (seed 42) and Stage 4 retrained on TBX11K.

Table 29: Self-attention vs. GAT task head on TBX11K, parameter-matched, identical Stage 1–3 weights, seed 42 (a separate run from Table 27). PG is *active_tb* only (164 pairs).

Head	Macro F1 ↑	PG ↑	Params
Self-attention (matched)	**0.906**	0.085	48,003
GAT (GRAPE)	0.900	**0.091**	43,587

Interpretation. Self-attention and the GAT land within seed variance of one another on both metrics, each winning narrowly on one of the two. Both are nonlinear readouts, and the self-attention head matches the GAT's performance without requiring a predefined adjacency, consistent with Appendix S's finding that the fixed co-occurrence graph is not responsible for the observed gain.

U Co-Occurrence Graph Robustness Under Distribution Shift

We test how robust the static training-label co-occurrence graph is to a shifted disease prevalence, institutional case mix, or label co-occurrence pattern. We resample the NIH training set to roughly double the prevalence of four findings (Infiltration $17.6\% \to 34.1\%$, Consolidation $4.1\% \to 7.6\%$, Pneumonia $1.3\% \to 2.5\%$, Effusion $11.8\% \to 22.6\%$), simulating a different institutional case mix, and rebuild the co-occurrence graph $\mathbf{A}$ from the resampled labels at the same threshold $\tau = 0.05$. The graph changes structurally, not just numerically: 12 edges added, 2 removed. We measure macro-F1 on the original (unshifted) held-out test set, since that is what a deployed model is actually judged against, under three conditions:

Table 30: Macro-F1 on the original NIH test set (4,500 images) after resampling the training set to raise the prevalence of four findings and rebuilding the graph at $\tau = 0.05$ (seed 42, single run).

Condition	Macro F1
Baseline (original graph and training data)	0.6002
Retrained under shift (new graph, Stage 4 retrained)	0.6321 (+3.19 pp)
Static-graph deployment (graph swapped only)	0.6002 (identical: graph unused)

V In-Domain Backbone Ablation

We test whether GRAPE's conclusions change with an in-domain foundation-model backbone instead of ImageNet-pretrained ResNet-50. Module A operates on the prototype similarity score matrix $\mathbf{S}$, not on backbone features directly, so swapping the backbone is expected to change the quality of the similarity scores Module A receives rather than what Module A does with them. The same holds for Modules B and C, which are defined over similarity maps derived from $\mathbf{S}$. We retrain the full 4-stage pipeline from scratch on TBX11K (seed 42, otherwise identical protocol) with the backbone replaced by TorchXRayVision's DenseNet-121 [5], pretrained on a pool of public chest X-ray datasets (an in-domain radiology foundation model, as opposed to ImageNet-pretrained ResNet-50), producing 1024-channel 7×7 feature maps in place of ResNet-50's 2048-channel 7×7 maps. F1 falls by 7.5 pp and PG rises by 5.8 pp at the same resolution (Table 31). With a single seed and no re-tuning, this is not a definitive ranking of backbones.

Table 31: Backbone ablation on TBX11K: full four-stage retraining (GNN+Unc.+VLM, seed 42, FP16; hyperparameters not re-tuned for DenseNet-121; single run). PG over the 207 `bbox_eval` pairs; both backbones yield 7×7 feature maps.

Backbone	Macro F1 ↑	PG ↑	Feature dim
ResNet-50 (ImageNet)	**0.9001**	0.0821	2048
TorchXRayVision DenseNet-121 (in-domain CXR)	0.8255	**0.1401**	1024

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state three specific claims, each scoped to the dataset/condition it was measured under — +13.8 pp F1 on TBX11K ($p = 0.0013$, 5 seeds; not established on NIH at $K = 14$, §4.3), 85% TP safety check with 16.4% FP at a selected operating point (with substantially lower TP at clinically realistic false-positive budgets on both datasets, §4.5), and 1-shot concept addition at low absolute Pointing Game values (evaluated for localization only, with its downstream cost to the pre-existing model reported in §4.6) — all supported by experiments in §4.2, §4.5, and §4.6. Limitations (null NIH F1, non-significant PG multi-seed, no graph-specific benefit established, simulated safety study as a disagreement proxy rather than validated clinician behavior, Module C's downstream interference) are explicitly disclosed in §4.5, §4.6, §5, and Table 3.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A dedicated Limitations paragraph in §5 covers: (1) low absolute Pointing Game scores due to 7×7 resolution and approximate NIH box placement (an in-domain backbone at the same resolution is compared in Appendix V); (2) Module B's sensitivity at clinically realistic false-positive budgets on both datasets (substantially lower than the default operating point, more severely on NIH, Appendix Q), the absence of a clinical user study, and the safety check being a controllable disagreement proxy rather than a validated model of clinician error; (3) Module C validated on 3 concepts only with full $K = 14$ LOO left for future work (its downstream interference cost is reported in §4.6); (4) NIH F1 null result and no graph-specific benefit being established (Appendix S); and (5) the static co-occurrence graph's robustness to distribution shift not being established (Appendix U). Training variance (TBX11K F1 std = 0.042) is also reported in §5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution

is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.

- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper makes no formal theoretical claims (no theorems, lemmas, or proofs). All contributions are architectural and empirical. Mathematical details (GAT attention derivation, loss functions, co-occurrence normalization) are fully specified in §3 and Appendix K.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Full hyperparameters (architecture, optimizer, learning rates, epochs, batch size, loss weights) are listed in Appendix B (Tables 7 and 8). The staged training algorithm is given in Appendix J. Dataset splits, evaluation protocols, and random seeds ($\{0, 1, 2, 3, 42\}$) are specified in §4.1 and §4.2. Code and training scripts are available at `https://github.com/KurbanIntelligenceLab/GRAPE` (§6).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same

dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code and training scripts are available at `https://github.com/KurbanIntelligenceLab/GRAPE` (§6). Both datasets used (TBX11K and NIH ChestX-ray14) are publicly available under their respective research licences; access instructions and citations are provided in §4.1 and the Broader Impact section. Hyperparameters and algorithmic details sufficient for re-implementation are in Appendix B and J.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: §4.1 specifies dataset splits and evaluation protocols. Appendix B provides all hyperparameters including optimizer (AdamW), learning rates, weight decay, batch size, epoch counts, and hardware (NVIDIA A100 80GB and Tesla P100 16GB). Threshold choices (τ, η) are justified via sensitivity sweeps (Tables 5 and 6).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Main results (Table 3) report mean $\pm$ std over 5 seeds with paired t-test p-values ($p = 0.0013$ for TBX11K F1; $p = 0.894$ for NIH F1; $p = 0.64$ for PG). Module B safety results include bootstrap 95% CIs ($B = 1000$ resamples). Standard deviations are computed across seeds.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The acknowledgments and Appendix B state that the main experiments use a single NVIDIA A100 80GB GPU with BF16 AMP, and the later ablations (Appendix S) a single Tesla P100 16GB GPU with FP16 AMP. Inference latency (GNN head: +1 ms over linear head at batch size 1) is reported in §4.3 and Table 12. Training epoch counts per stage and dataset are presented in Table 8.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper uses only publicly released, de-identified datasets (TBX11K, NIH ChestX-ray14) under their respective research licenses; no new human-subject data were collected. An ethics statement is included in the Broader Impact section. The system is a research prototype not deployed in clinical settings.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The Broader Impact section discusses positive impacts (traceable clinical predictions, reduced harmful feedback amplification, lower adoption barriers) and negative impacts (over-reliance on plausible-but-unfaithful explanations, unaudited subgroup performance, potential demographic bias in prototype atlases). Mitigation recommendations are included.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release new datasets or high-risk generative models. GRAPE is a discriminative medical image classifier trained on de-identified public data; no safeguards beyond standard research-use licensing are required. Clinical deployment is explicitly out of scope.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: TBX11K [12] (research licence) and NIH ChestX-ray14 [23] (NIH public-access licence) are cited and their licences described in the Broader Impact section. BioViL-T [1] is used as a frozen encoder and is cited accordingly. All baseline implementations (CBM, PIPNet, CSR) are cited with their original papers.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper does not release new datasets. The GRAPE training code is released at `https://github.com/KurbanIntelligenceLab/GRAPE`, documented in its README.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human-subject research. The safety check evaluation (§4.5) is a simulated miscorrection study on existing annotated data; no human participants were recruited.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No human subjects were involved. All experiments use retrospective, de-identified, publicly released medical imaging datasets (TBX11K, NIH ChestX-ray14), which do not require IRB approval for secondary analysis under their respective release terms.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not used as a component of the GRAPE architecture or experimental pipeline. BioViL-T is a domain-specific vision-language model used as a frozen text encoder for prototype anchoring, not an LLM in the generative sense; its use is already cited and described in §3.4.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.